Technische Universität München

Department of Mathematics

Bachelor's Thesis

Computational Study of Equivalence of Graphical Models with Groupwise Equal Error variances

Leonard Waldmann

Supervisor: Prof. Dr. Mathias Drton

Advisor: Dr. Roser Homs Pons

Submission Date: 31st March 2022

I assure the single handed composition of this bachelor's thesis only supported by declared resources.

Garching,

Zusammenfassung

Die Kanten eines gerichteten azyklischen Graphen (DAG) können als Ursache-Wirkung-Beziehung von Eltern- zu Kindknoten interpretiert werden. Lineare Strukturgleichungsmodelle (LSEM) sind Mengen von Kovarianzmatrizen, deren Verteilungen diese qualitative Interpretation quantifizieren und jeden Knoten als lineare Funktion aller Elternknoten mit einem unabhängigen additiven Fehlerterm darstellen. Es wurde bereits gezeigt, dass, ausgehend von dem LSEM eines unbekannten DAG, nur die Markov-Äquivalenzklasse des ursprünglichen Graphen identifiziert werden kann. Unter der zusätzlichen Annahme, dass die Fehlerterme identische Varianzen haben, kann der Graph jedoch eindeutig identifiziert werden. Das Ziel der vorliegenden Arbeit ist die Erkundung dieser sogenanten Kovarianz-Äquivalenzklassen im allgemeinen Fall, also unter beliebiger Aufteilung der Knoten in Gruppen mit identischen Fehlervarianzen. Das Hauptergebnis dieser Arbeit ist die Berechnung der Kovarianz-Äquivalenzklassen von DAGs mit drei und vier Knoten unter beliebigen gruppenweise identischen Fehlervarianzen mithilfe der Computer Algebra Software Macaulay2 und Maple. Basierend auf diesen Ergebnissen wird eine Vermutung für eine beliebige Anzahl an Knoten aufgestellt, welche die beiden bekannten Fälle impliziert und mit den rechnerischen Ergebnissen übereinstimmt. Unter der Annahme, dass die Vermutung zutrifft, werden die Größen dieser neuen Kovarianz-Äquivalenzklassen bis zu Graphen mit sechs Knoten untersucht. Außerdem wird der Algorithmus für die Berechnung von gerichteter zyklischer Graphen erweitert, es kann aber nicht garantiert werden, dass diese Ergebnisse den Kovarianz-Äquivalenzklassen der gerichteten Graphen entsprechen. Der gesamte Code ist für weiterführende Untersuchungen verfügbar.

Summary

The edges of a directed acyclic graph (DAG) can be interpreted as cause-effect relations from parent to child node. Linear structural equation models (LSEM) are sets of matrices of distributions that quantify these cause-effect relations by describing a node through a linear function of its parents and some additive independent noise term. Given the LSEM of an unknown DAG, previous results have shown that the underlying DAG can only be identified up to the well-known Markov equivalence classes. Under the additional assumption that all error terms have identical variance, it has been proven that the graph can be identified uniquely. The goal of this thesis is to explore these so-called covariance equivalence classes in the general setting of an arbitrary partition of the nodes into groups with equal error variances. The main result of this thesis is the computation of covariance equivalence classes of DAGs with three and four nodes under arbitrary groups of equal error variances with the aid of the computational algebra software Macaulay2 and Maple. Based on these results, a conjecture on the case of arbitrary number of nodes that entails the two known equal error variance settings and explains the computational results is stated. Assuming that the conjecture holds true, the distribution of the sizes of covariance equivalence classes of DAGs up to six nodes is explored. Moreover, the algorithm is extended to allow cyclic graphs, however, it cannot be ensured that these results correspond to covariance equivalence classes of directed cyclic graphs. All code is available for further investigations.

Contents

1   Introduction

Finding cause-effect relations in multivariate non-interventional data is of great interest in practice and is used inter alia for medical diagnosis [17] or environmental risk assessment [13]. Naturally, cause-effect relations can be depicted by a directed graph where each node corresponds to an event and each edge to a causation from parent to child. We further assume that the graph has no cycles so these relations can also be interpreted as being subsequent in time. An example is shown in Figure 1.
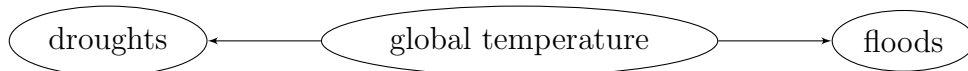

Figure 1: Cause-effect graph of global temperature, droughts, and floods.

When associating each of the n nodes with a random variable X_i, these qualitative graphs generate quantitative structural equation models that consist of all multivariate distributions $X = (X_1, ..., X_n)$ such that each random variable can be modelled as a function of its parents and some independent noise term. Throughout this thesis, we will work with the restricted linear structural equation models which further assume that the functions are linear with coefficients λ_{jk} for the edge $j \to k$ and the noise terms ε_j are additive. In particular, we define the model as the set of covariance matrices of X such that

$$X_j = \sum_{i \in \mathrm{pa}(j)} \lambda_{ij} X_i + \varepsilon_j, \quad j = 1, ..., n. \tag{1}$$

Now, our question of interest is the following; given a linear structural equation model for some unknown directed acyclic graph G, can we identify the underlying causal structure, i.e. the graph G? Under additional mild assumption, answers to this question give results on whether we can recover the causal structure given just a single covariance matrix that could e.g. be the sample covariance matrix from observational data in practice.

If different graphs have identical models, we cannot identify the causal structure from the model. We call graphs with this property covariance equivalent. In general, different graphs can be covariance equivalent and only the well-known Markov equivalence classes can be identified [25]. A recent advance has shown that it is possible to break the Markov equivalence classes and uniquely identify the underlying graph by introducing the additional assumption of equal error variances [20, 1]. These results give rise to the leading question of this thesis; what characterizes covariance equivalence classes in the general case of arbitrary groups of nodes with equal error variances?

Section 2 gives a functional and graphical description of the linear structural equation model. Section 3 reviews the two known error variance settings and rephrases them in a new notation. Section 4 utilizes computational algebra to calculate covariance equivalence classes under arbitrary groups of equal error variances, applies these tools in Macaulay2 and Maple, and explores the results on directed acyclic graphs with three and four nodes. Finally, Section 5 states a conjecture that entails the known settings and explains all computational results, explores equivalence class sizes of graphs up to six nodes under the conjecture, and shows results from an extension of the implementation to any directed graph without loops that, however, cannot guaranteed to be interpreted identically.

2   Preliminaries on graphical modelling

Directed acyclic graphs can be interpreted as cause-effect relations from parent to child node. In this section, we will review a rigorous functional and graphical description of this notion, the first through equations and the latter through conditional independence statements. The functional version is also called linear structural equation model and is the main model of consideration in this thesis. Finally, we will see that these descriptions are very closely related.

2.1   Linear structural equation model

The definitions and propositions of this section follow the review from Drton [3]. We start by introducing some common language on directed graphs used throughout this thesis. A *directed graph* G is a tuple (V, D) of *vertices* V and *directed edges* D where V is finite and $D \subset V \times V$. Without loss of generality, we will assume that the nodes are labeled from 1 to $|V|$, i.e. $V = \{1, ..., |V|\} =: [|V|]$. Instead of writing $(v, w) \in D$, we also write $v \to w$. We denote the *parents* of a node $v \in V$ with $\mathrm{pa}(v) := \{w \in V \mid (w, v) \in D\}$. A *directed path* of length $l \in \mathbb{N}$ is a tuple of vertices $(v_1, ..., v_l)$ such that $(v_i, v_{i+1}) \in D$ for $i \in [l-1]$. A directed graph is *acyclic* if there exists no directed path of any length $l \in \mathbb{N}$ such that $v_1 = v_l$. In the following, we will almost exclusively work with directed acyclic graphs (DAG). The *descendants* of a node v, denoted by $\mathrm{de}(v)$, are all nodes $w \in V$ such that there exists a directed path from v to w and the *non descendants* denoted by $\mathrm{nd}(v)$ are the nodes $w \in V \setminus \{v\}$ such that there does not exist directed path from v to w. An important property of a directed acyclic graph $G = ([n], D)$ is that we can always permute the labels of the vertices such that $v \in \mathrm{pa}(w)$ implies $v < w$ for any $v, w \in [n]$. This is also called *topological ordering* [24, Thm. 13.2.10].

Now, let $G := (V, D)$ be a DAG with n vertices, X be a n-dimensional random vector and $\varepsilon = (\varepsilon_1, ..., \varepsilon_n)$ some pairwise independent noise terms with $\mathbb{E}(\varepsilon) = 0$. Assume that the equation system

$$X_j = \sum_{i \in \mathrm{pa}(j)} \lambda_{ij} X_i + \varepsilon_j, \quad j \in 1, \ldots n \tag{2}$$

holds for some $\Lambda = (\lambda_{ij}) \in \mathbb{R}^{n \times n}$ and $\Omega := \mathrm{Var}[\varepsilon] = \mathrm{diag}(\sigma_1^2, ..., \sigma_n^2)$ with all $\sigma_i^2 > 0$. The coefficient λ_{ij} quantifies the cause-effect relation from node i to node j, so we assume Λ to be in the set

$$\mathbb{R}^D := \{\Lambda \in \mathbb{R}^{n \times n} \mid \lambda_{ij} = 0 \text{ if } i \to j \notin D\}. \tag{3}$$

We define the set of all possible Ω matrices as

$$DP_n := \{\Omega \in \mathbb{R}^{n \times n} \mid \omega_{ij} = 0 \text{ for } i \neq j, \ \omega_{ii} > 0 \text{ for } i \in [n]\} \subset PD_n \tag{4}$$

where PD_n denotes the set of all positive definite matrices in $\mathbb{R}^{n \times n}$. Originating from Equation (2), a straight-forward way to define the linear structural equation model of the graph G would be the set of all distributions of X such that X satisfies Equation (2) for some $\Lambda \in \mathbb{R}^D$ and $\Omega \in DP_n$. For the sake of argument, let us call this set of distributions *actual model* and let us call two graphs that have identical actual models *actually equivalent*. Our interest in this thesis, however, only lies in the second moment of X, i.e. the variance $\Sigma := \mathrm{Var}[X]$. We do not want to incorporate any higher moments

and without loss of generality we can assume that the first moment of X is zero since we can always center the data at zero with a linear transformation. This leaves us with defining the linear structural equation model as the set of covariance matrices of the distributions in the actual model. Our notion of equivalence of graphs will thus only refer to equality of some covariance matrices. To emphasize that we are only looking at the second moment, we will call this *covariance equivalence*. In general, covariance equivalence will not imply actual equivalence. However, if we confine us to cases where the noise terms have specific distributions such that the distribution of X is fully determined by its first and second moment, the notions of covariance equivalence and actual equivalence are identical. The most important case where this happens are Gaussian noise terms. Then, X is Gaussian and a Gaussian distribution is fully determined by its mean and covariance. In this fashion, the results of covariance equivalence can be applied to any distribution, but make particularly sense if we further require that X has a distribution that is fully determined by its first two moments. The remainder of this section makes the definition of the linear structural equation model rigorous.

Writing Equation (2) in matrix notation, we obtain

$$X = \Lambda^T X + \varepsilon. \tag{5}$$

Let I denote the identity matrix, then we rearrange to $(I - \Lambda)^T X = \varepsilon$ and compute the variance on both sides

$$(I - \Lambda)^T \mathrm{Var}[X](I - \Lambda) = \mathrm{Var}[\varepsilon]. \tag{6}$$

Now define $\Sigma := \mathrm{Var}[X]$ and we obtain

$$(I - \Lambda)^T \Sigma (I - \Lambda) = \Omega. \tag{7}$$

This is the central equation of the linear structural equation model of G as it states the only equation that a covariance matrix Σ needs to fulfill in order to be in the model. Since we assume G to be acyclic, we can even give Equation (7) with explicit Σ.

Proposition 2.1. *[3, p. 5] If graph $G = (V, D)$ is acyclic, then $(I - \Lambda)$ is invertible for all $\Lambda \in \mathbb{R}^D$.*

Proof. Since G is acyclic, there exists a permutation $\sigma \in S_n$ of the labels of G such that the permuted labels are in topological order. By permuting both rows and columns of $(I - \Lambda)$ with σ it is clear that the resulting matrix is an upper triangular matrix with ones in the diagonal. We obtain $\det(I - \Lambda) = 1$. $\qquad \square$

We rearrange Equation (7) to

$$\Sigma = (I - \Lambda)^{-T} \Omega (I - \Lambda)^{-1} \tag{8}$$

where the $-T$ in the exponent denotes the inverse of the transposed matrix. Now, we use this explicit equation and the domains of Λ and Ω from Equation (3) and Equation (4) as definition of the linear structural equation model.

Definition 2.2. The *linear structural equation model* (LSEM) of a directed acyclic graph $G = (V, D)$ with n nodes is defined as

$$\mathcal{M}(G) = \{(I - \Lambda)^{-T} \Omega (I - \Lambda)^{-1} \mid \Lambda \in \mathbb{R}^D, \Omega \in DP_n\}. \tag{9}$$

We might just say *model* instead of linear structural equation model if the context is clear. The following proposition shows the model actually contains covariance matrices.

Proposition 2.3. *Let G be a DAG with n nodes. Then, $\mathcal{M}(G) \subset PD_n$.*

Proof. Let $\Sigma = (I - \Lambda)^{-T}\Omega(I - \Lambda)^{-1} \in \mathcal{M}(G)$ with $\Lambda \in \mathbb{R}^D, \Omega \in DP_n$. We use that the hermitian matrix M is positive definite if and only if there exists an invertible $B \in \mathbb{C}^{n \times n}$ such that $M = B^T B$ which is a result closely related to the Cholesky factorization of positive definite matrices [9, 7.2.7 / 7.2.P9]. Since $\Omega \in PD_n$, there exists an invertible $B \in \mathbb{C}^{n \times n}$ such that $\Omega = B^T B$. Define $A := B(I - \Lambda)^{-1}$ then A invertible with $\Sigma = (I - \Lambda)^{-T}B^T B(I - \Lambda)^{-1} = A^T A$, so Σ positive definite. $\qquad \square$

We emphasize that the definition of linear structural equation models directly yields a parametrization. Let us have a closer look.

Proposition 2.4. *[3, Prop. 2.1] Let $G = (V, E)$ be a DAG with n nodes. Define*

$$\phi_G : \mathbb{R}^D \times \{\mathrm{diag}(d_1, ..., d_n) \in \mathbb{R}^{n \times n} \mid d_i \in \mathbb{R}\}, \ (\Lambda, \Omega) \mapsto (I - \Lambda)^{-T}\Omega(I - \Lambda)^{-1}. \qquad (10)$$

Then ϕ_G is a well-defined polynomial map that parametrizes the linear structural equation model of G, i.e.

$$\mathcal{M}(G) = \phi_G(\mathbb{R}^D, DP_n). \qquad (11)$$

Proof. That ϕ_G is well-defined and that Equation (11) holds follows from Proposition 2.1 and Definition 2.2. By Cramer's rule for matrix inversion we have that the inverse of $(I - \Lambda)$ is $(I - \Lambda)^{-1} = \det(I - \Lambda)^{-1}\mathrm{adj}(I - \Lambda)$ where $\mathrm{adj}(\cdot)$ denotes the adjugate matrix. In Proposition 2.1, we have already seen that $\det(I - \Lambda)^{-1} = 1$. Since the adjugate matrix is polynomial in the entries of its original matrix by definition, we conclude that $(I - \Lambda)^{-1}$ is polynomial in Λ. Now clearly, ϕ_G polynomial in (Λ, Ω). $\qquad \square$

The fact that ϕ_G is a polynomial map that parametrizes the model is an important fact and will allow us to use algebraic tools in Section 4.1. Now, we define our notion of equivalence of graphs.

Definition 2.5. Two DAGs G and G' with identical vertices are *covariance equivalent* if they have identical LSEMs, i.e. $\mathcal{M}(G) = \mathcal{M}(G')$.

We close this section with two remarks pointing at two interesting topics on LSEM beyond the scope of this thesis.

Remark 2.6 (Parameter identifiability)**.** An interesting question is whether we can uniquely identify the strengths of the cause-effect relations and the error variances, i.e. the entries of the matrices Λ and Ω, from the joint distribution given the DAG G. If G is a DAG as in our setting, it turns out that we can always identify both Λ and Ω since ϕ_G is invertible [24, Thm. 16.2.1].

Remark 2.7 (More general graphs)**.** Instead of considering a DAG, we could also consider more general graphs, e.g. digraphs that allow cycles or mixed graphs. Changing the graph changes the interpretation of dependence [24, p. 283], e.g. our cause-effect relation in DAGs would need to incorporate the idea of feedback loops for cyclic directed graphs.

The undirected edges of the mixed graph correspond to correlated noise terms and are interpreted as influences from unobserved, also called latent, variables. We note, that parameter identifiability in the sense of the previous remark cannot be guaranteed when using cyclic or mixed graphs [24, Thm. 16.2.1]. This gives rise to other notions of parameter identifiability that are reviewed by e.g. Sullivant [24, Chapter 16]. Readers further interested in mixed graphs that allow cycles may consult the review of Drton [3].

2.2   Conditional independence and Markov properties

Conditional independence statements are easy interpretable and important statements in statistics [24, Chapter 4] and in this section, we will see how we can describe the LSEM from Section 2.1 through a set of conditional independence statements. We will closely follow the argumentation and definitions of Sullivant [24, Chapter 4 & 13] and Neapolitan [19, Chapter 2]. Throughout this section, X will be a random vector of dimension n that takes values in $\mathcal{X} = \prod_{i=1}^{n} \mathcal{X}_i$. Moreover, we assume that X has a joint density f with respect to a probability measure ν on $\mathcal{X}$. For $A \subset [n]$, we denote the subvector of X with indices in A as X_A. We recall the definition of conditional independence for random vectors with densities as stated by Sullivant [24, Chapter 4.1].

Definition 2.8. Let $A, B, C \subset [n]$ be pairwise disjoint and $x_B \in \mathcal{X}_B$.

(a) The *marginal density* $f_A(x_A)$ of X_A is

$$f_A\left(x_A\right) := \int_{\mathcal{X}_{[n]\setminus A}} f\left(x_A, x_{[n]\setminus A}\right) d\nu_{[n]\setminus A}\left(x_{[n]\setminus A}\right) \quad \text{for } x_A \in \mathcal{X}_A. \tag{12}$$

(b) The *conditional density* of X_A given $X_B = x_B$ is

$$f_{A|B}\left(x_A \mid x_B\right) := \begin{cases} \frac{f_{A\cup B}(x_A, x_B)}{f_B(x_B)} & \text{if } f_B\left(x_B\right) > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

(c) The random vector X_A is *conditionally independent* of X_B given X_C if and only if

$$f_{A\cup B|C}\left(x_A, x_B \mid x_C\right) = f_{A|C}\left(x_A \mid x_C\right) \cdot f_{B|C}\left(x_B \mid x_C\right). \tag{14}$$

In this case, we write $X_A \perp\!\!\!\perp X_B \mid X_C$ or just $A \perp\!\!\!\perp B \mid C$ if X is clear from context.

Intuitively, $X_A \perp\!\!\!\perp X_B \mid X_C$ implies that "given X_C, knowing X_B does not give any information about X_A" [24, p. 71]. Since we interpret the edges of a DAG $G = (V, D)$ as cause-effect relations, we would expect that any node $v \in V$ is determined by its parents $\mathrm{pa}(v)$ and has influence on precisely all its descendants $\mathrm{de}(v)$. Phrased with conditional independence, given the parents of v, knowing the remaining non-descendants of v gives us no information about v, i.e. $v \perp\!\!\!\perp \mathrm{nd}(v) \setminus \mathrm{pa}(v) \mid \mathrm{pa}(v)$. Hence, we could also describe distributions that follow some kind of cause-effect relation of G through the following conditional independence statements.

Definition 2.9. Let $G = (V, D)$ be a DAG. The *local Markov property* of G is the set of all conditional independence statements

$$X_v \perp\!\!\!\perp X_{\mathrm{nd}(v)\backslash\mathrm{pa}(v)} \mid X_{\mathrm{pa}(v)} \tag{15}$$

where $v \in V$. If a distribution ν satisfies all of the above conditional independence statements, we say ν satisfies the local Markov property of G.

The set of distributions that satisfy the local Markov property of a DAG G is also called *directed graphical model* of G. Given the local Markov property of G, we ask if there are any more conditional independence statements that a distribution has to satisfy if it satisfies the local Markov property of G. It turns out that even for simple examples, there are. Take the graph $G = 1 \to 2 \to 3$, then the local Markov property of G only contains $3 \perp\!\!\!\perp 1 \mid 2$. From the definition of conditional independence it is clear that this also implies $1 \perp\!\!\!\perp 3 \mid 2$ but the latter is not in the local Markov property. This symmetry of conditional independence statements in its first two arguments is one of four straight-forward implications from the definition of conditional independence, also called *conditional independence axioms* [24, Prop. 4.1.4]. The fact that there are additional conditional independence statements implied by the local Markov property motivates the following definition from Neapolitan [19, Def. 2.1].

Definition 2.10. Let G be a DAG and $A, B, C \subset [n]$. We say that G *entails* $A \perp\!\!\!\perp B \mid C$ if $A \perp\!\!\!\perp B \mid C$ holds for all distributions that satisfy the local Markov property of G.

As stated by Sullivant [24, p. 71], it is a very challenging problem to find the set of all implied conditional independence statements in general and it is known that there exist no finite set of rules from which all implied conditional independence statements can be deduced in general [23]. However, it turns out that our case is endowed with a graphical property that perfectly describes the set of entailed conditional independencies. Before we can state this property exactly as Sullivant [24, Def. 13.1.7], we need some more notation. An *undirected path* in a directed graph $G = (V, D)$ is a tuple $(v_1, ..., v_l)$ with $v_i \in V$ such that (v_i, v_{i+1}) or (v_{i+1}, v_i) for all $i \in [l-1]$ in D. The *ancestors* of a node $v \in V$ denoted by $\mathrm{an}(v)$ are all nodes $w \in V$ such that there exists a directed path from w to v.

Definition 2.11. Two nodes v and w in a directed acyclic graph G are *d-connected* given a set $C \subseteq V \backslash \{v, w\}$ if there is an undirected path π from v to w such that

   (i) all colliders on π are in $C \cup \mathrm{an}(C)$ and

   (ii) no non-collider on π is in C.

Let $A, B, C \subseteq V$ be pairwise disjoint with A and B nonempty, then C *d-separates* A and B if no pair of nodes $a \in A$ and $b \in B$ are d-connected given C. If A and B are d-separated by C we denote $A \perp_d B \mid C$ and if $C = \emptyset$ we denote $A \perp_d B$. If A, B or C are single element sets, we omit the curly braces.

Example 2.12. We want to list all d-separations of the DAGs $G_1 = 1 \to 2 \to 3$ and $G_2 = 1 \to 2 \leftarrow 3$. By the definition of d-separation, it immediately follows that d-separation is symmetric in its first two arguments and that adjacent nodes are always

d-connected. Hence, we just need to check whether the only undirected path from 1 to 3, i.e. $\pi = (1, 2, 3)$, d-connects 1 and 3 given 2 or given the empty set. In G_1, it holds that 1 and 3 are d-connected given $\emptyset$ but 1 and 3 are not d-connected given 2. Hence, all d-separations of G_1 are $\{1 \perp_d 3 \mid 2, \ 3 \perp_d 1 \mid 2\}$. In G_2, it holds that 1 and 3 are d-connected given 2 but 1 and 3 are not d-connected given $\emptyset$. Hence, all d-separations of G_2 are $\{1 \perp_d 3, \ 3 \perp_d 1\}$.

Now, we define the set of all conditional independence statements that can be characterized by d-separation and state the crucial observation that gives d-separation its importance.

Definition 2.13. Let G be a DAG. The *global Markov property* of G is the set of all conditional independence statements that are characterized by d-separation, i.e. all

$$X_A \perp\!\!\!\perp X_B \mid X_C \quad \text{such that} \quad A \perp_d B \mid C \tag{16}$$

with $A, B, C \subset [n]$. If a distribution ν satisfies all of the above conditional independence statements, we say ν satisfies the global Markov property of G.

Theorem 2.14. *Let G be a DAG with n nodes. Then the following hold:*

(i) *[14, §3.2.2] A distribution satisfies the local Markov property of G if and only if it satisfies the global Markov property of G.*

(ii) *[24, Prop. 13.1.12] For all $A, B, C \subset [n]$ such that A and B are not d-separated by C, there exists a distribution that satisfies the global Markov property of G but not $X_A \perp\!\!\!\perp X_B \mid X_C$.*

Clearly, this theorem implies that the set of conditional independencies entailed by G is precisely the global Markov property. Statement (ii) from the theorem above is also referred to as *completeness* of the global Markov property. The theorem yields that the d-separations of the global Markov property give an explicit description of the implicit conditional independence statements of the local Markov property and perfectly associate them with graphical properties.

Finally, we establish the connection of the conditional independence statements of the Markov properties with the LSEM given in Section 2.1. We say the density f *factorizes* as the product of conditionals with respect to G [5, Eq. 3.3.3], if

$$f(x) = \prod_{i=1}^{n} f_i(x_i \mid x_{\text{pa}(i)}). \tag{17}$$

It turns out, that this factorization is equivalent to the local Markov property.

Theorem 2.15 (Recursive factorization). *[24, Thm. 13.2.10] A probability density factorizes with respect to a DAG G if and only if it satisfies the local Markov property of G.*

The recursive factorization theorem yields that each node can be expressed through a density conditioned on its parental nodes that can be interpreted as prior variables, so directed graphical models are also called *Bayesian networks* [24, p. 296]. Specifically in the Gaussian case, we can establish a connection between factorization and the LSEM of Section 2.1.

Proposition 2.16. *[5, Prop. 3.3.12] A Gaussian density factorizes with respect to a DAG G if and only if its covariance matrix is in the LSEM $\mathcal{M}(G)$.*

Now, we can use this observation to connect the Markov properties with the LSEM.

Corollary 2.17. *Let G be a DAG. The set of covariance matrices of Gaussian distributions that satisfy the local Markov property of G equals $\mathcal{M}(G)$.*

Proof. This directly follows from Theorem 2.15 and Proposition 2.16.   □

In summary, we have found a convenient way to describe our LSEM from Section 2.1 through conditional independence statements that can perfectly be associated to the graphical property of d-separation.

3 Covariance equivalence

After giving a functional and graphical description of the LSEM in the last section, we will start this section by reviewing two known results on recovering causal structures from LSEMs under two different assumptions. In the setting with no additional assumptions, Verma and Pearl proved that only the well-known Markov equivalence classes of graphs can be recovered [25]. However, when additionally requiring all error variances to be equal, Peters and Buehlmann showed that the causal structure can be identified uniquely. [20, 1]. After introducing some additional notation on equal error variance assumptions, these results will motivate the leading question of this thesis.

3.1 Arbitrary error variances: Markov equivalence classes

In Section 2.2, we have described the LSEM of Section 2.1 through the conditional independence statements in the local or global Markov property, cf. Corollary 2.17. Now, it is intuitive to ask whether two different DAGs always have different global Markov properties or, equivalently, whether they have different d-separations. This is false in general.

Example 3.1. Let $G_3 = 1 \leftarrow 2 \leftarrow 3$. As in Example 2.12, we only need to check whether the undirected path $\pi = (1, 2, 3)$ d-connects 1 and 3 given 2 or given $\emptyset$. We see that π d-connects 1 and 3 given $\emptyset$ but not given 2, as 2 is not a collider. Hence, the global Markov property of G_3 is $\{1 \perp\!\!\!\perp 3 \mid 2, \ 3 \perp\!\!\!\perp 1 \mid 2\}$. Taking graph $G_1 = 1 \rightarrow 2 \rightarrow 3$ from Example 2.12, we see that G_1 and G_3 have identical Markov properties although they are not the same graphs.

This gives rise to the following definition.

Definition 3.2. Two DAGs G and G' are *Markov equivalent* if they have identical global Markov properties.

To check whether two graphs are Markov equivalent, we could compare all their d-separations. However, there is a more convenient graphical description of Markov equivalence that is well-known and has first been proven by Verma and Pearl [25]. We need the following definition.

Definition 3.3. Let $G = (V, D)$ be a directed graph.

(a) The *skeleton* of G is the undirected graph $(V, \{\{a, b\} \mid (a, b) \in D\})$ that removes the directions from the edges of G.

(b) An *unshielded collider* of G is a triple of nodes (a, c, b) such that $(a, c), (b, c) \in D$ but $(a, b), (b, a) \notin D$.

Now we can state the observation from Verma and Pearl [25].

Theorem 3.4. *[25, Thm. 1] Let G, G' be DAGs. Then, G and G' are Markov equivalent if and only if they have identical skeleton and identical unshielded colliders.*

Now, Markov equivalence can conveniently be checked in polynomial time [19, p. 91] through the graphical properties of skeleton and unshielded colliders. Take the graphs G_1, G_2, and G_3 from Example 2.12 and Example 3.1, then we immediately see that G_1 and G_3 are Markov equivalent but G_2 is not Markov equivalent to G_1 and G_3 since G_2 has an unshielded collider in node 2. Clearly, two Markov equivalent graphs cannot be differentiated based on global Markov properties or, equivalently, their LSEMs, cf. Corollary 2.17. This is an undesired behaviour as different graphs encode different cause-effect interpretations. The graphs that are Markov equivalent to one another form *Markov equivalence classes* and we say that graphs can only be recovered up to Markov equivalence classes in the current setting.

Remark 3.5. In practice, given some observational data of a directed graphical model, we can estimate the joint distribution and in particular its covariance matrix from the data [3, p. 7] and want to recover all causal structures, i.e. DAGs, that could have generated this distribution. This setting is slightly different in the sense that we only have a single covariance matrix and not the whole model to recover the graph from. The problem is the following: if two graphs are Markov equivalent, their LSEMs are identical so they generate the same set of covariance matrices, but if two graphs are not Markov equivalent, their LSEMs can still intersect and covariance matrices in the intersection of the models could again be generated by both graphs. To guarantee recovery up to Markov equivalence classes from single covariance matrices, we need an additional assumption called *faithfulness*. Faithfulness is defined as the reverse implication of the global Markov property, i.e. that the conditional independence statements of a distribution that satisfies the global Markov property of G are only those characterized by the d-separations of G [19, Chapter 2]. Faithfulness intuitively arises from our cause-effect interpretation of DAGs and Zhang and Sprites state that faithfulness is "usually made explicit —and when not, [...][is] usually implicit—"[26, p. 240] in our framework. Interested readers may consult the book of Neapolitan for further information [19, Chapter 2]. Methods for recovering the Markov equivalence class from a single distribution assuming faithfulness can be found in the review of Drton and Maathuis [4, Chapter 4]. These methods usually try to recover a representative of the Markov equivalence class that contains $v \to w$ if and only if all graphs in the equivalence class contain $v \to w$ and that contains the undirected edge $v \leftrightarrow w$ if and only if both $v \to w$ and $w \to v$ are present in graphs in the equivalence class [4, Chapter 4].

3.2   Equal error variances: Full identifiability

It is natural to ask whether we can impose additional assumptions to break the undesired Markov equivalence classes of the previous section and uniquely identify the causal structure. It turns out that this is possible if we further require that the error variances are equal [20, 1]. Whilst Peters and Bühlmann [20] first gave a proof by contradiction, Chen, Drton, and Wang [1] observed that the result is implied by an ordering of some conditional variances and provided a fast algorithm for estimating this ordering. We review the result with the notation of the paper from Chen, Drton, and Wang but with transposed definition of the coefficient matrix. This only has notational influence and the transposed version aligns with Section 2.1 of this thesis. Let X be a centered random

vector of dimension p. If the equation system

$$X_j = \sum_{k \neq j} \beta_{kj} X_k + \varepsilon_j, \quad j = 1, \ldots, p \tag{18}$$

holds for some $B = (\beta_{ij}) \in \mathbb{R}^{n \times n}$ and some $\sigma^2 = \text{Var}[\varepsilon_j]$ for all $j \in [p]$, we denote $X \sim (B, \sigma^2)$. The causal structure of this equation system can be interpreted as a graph $G(B)$ that has p vertices and the edge $j \to k$ if and only if $\beta_{jk} \neq 0$ for $j, k \in [p]$. The paper requires $G(B)$ to be a DAG. The main result of Peters and Bühlmann [20, Thm. 1] as stated by Chen, Drton, and Wang [1, Thm. 1] is the following.

Theorem 3.6. *[3, 20, Thm 1.] Let $X \sim (B_X, \sigma_X^2)$ and $Y \sim (B_Y, \sigma_Y^2)$ with both $\mathcal{G}(B_X)$ and $\mathcal{G}(B_Y)$ directed and acyclic. If $\text{Var}[X] = \text{Var}[Y]$, then $\mathcal{G}(B_X) = \mathcal{G}(B_Y)$, $B_X = B_Y$, and $\sigma_X^2 = \sigma_Y^2$.*

Theorem 3.6 uniquely identifies both the causal structure, i.e. the edges of the graph, and the strengths of the cause-effect relations, i.e. the entries of matrix B, from a single covariance matrix. This property is called *full identifiability* by Peters and Bühlmann [20]. It is clear that the notion $X \sim (B, \sigma^2)$ with $G(B)$ being a DAG is equivalent to saying that $\text{Var}[X]$ is in a LSEM as described in Section 2.1 with the two following additional assumptions; (i) all error variances σ_j^2 are identical and (ii) $\beta_{jk} = 0$ only if $j \to k \notin D$.

Remark 3.7. Theorem 3.6 directly gives a result on recovering the causal structure from single covariance matrices without needing the faithfulness assumption, cf. Remark 3.5. However, assuming (ii) implies a weak form of faithfulness called *causal minimality* [20, Problem 3]. In particular, assumption (ii) breaks model subset relations for any DAGs $G = (V, D)$ and $G' = (V, D')$ with $D \subset D'$, as we clearly have $\mathcal{M}(G) \subset \mathcal{M}(G')$ by setting the edge coefficients of $D' \backslash D$ to 0. As Peters and Bühlmann argue, causal minimality "is a natural condition and in accordance with the intuitive understanding of a causal influence between the variables" [20, Problem 3]. Thus, it seems like an intuitive assumption for retrieving the causal structure from single covariance matrices.

3.3 Arbitrary groups of equal error variances: A case to explore

We have reviewed that, given a LSEM, we can only recover equivalence classes of DAGs when all error variances are arbitrary but we can recover the unique graph if all error variances are equal. The question of what happens in the more general case when assuming that there is an arbitrary partition of the nodes into groups with equal error variances arises naturally. In the following, we will refer to this setting as arbitrary groups of equal error variances or groupwise equal error variances. The sole purpose of this section is to rephrase the previous results as statements about covariance equivalence with the aid of some new notation and properly formulate the upper question, the leading question of this thesis. All further investigations are content of Section 4 and Section 5.

For a DAG $G = (V, D)$, we can fully describe arbitrary groups of equal error variances by a partition $\mathcal{P}$ of the vertices V, where for $i, j \in V$ holds

$$(\exists P \in \mathcal{P} : \{i, j\} \subset P) \implies \text{Var}[\varepsilon_i] = \text{Var}[\varepsilon_j]. \tag{19}$$

Definition 3.8. We call a partition of the vertex set of a DAG G an *equal error variance partition* of G.

If it is clear from the context, we might just say equal variance partition. If we talk about G with the additional assumption of a variance partition $\mathcal{P}$ of G we say G under $\mathcal{P}$. By assuming some variance partition $\mathcal{P}$, we put additional constraints on the diagonal entries of DP_n and obtain

$$DP_n^{\mathcal{P}} := \{A = (a_{ij}) \in DP_n \mid \forall i, j \in [n] : (\exists P \in \mathcal{P} : \{i, j\} \subset P \Rightarrow a_{ii} = a_{jj})\}. \qquad (20)$$

Now we define the LSEM of G under $\mathcal{P}$ as follows.

Definition 3.9. Let G be a DAG with n nodes and $\mathcal{P}$ be an equal variance partition. Then the *linear structural equation model* (LSEM) of G under $\mathcal{P}$ is defined as

$$\mathcal{M}^{\mathcal{P}}(G) := \{(I - \Lambda)^{-T} \Omega (I - \Lambda)^{-1} \mid \Lambda \in \mathbb{R}^D, \Omega \in DP_n^{\mathcal{P}}\}. \qquad (21)$$

With analogous proof as in Section 2.1, we obtain the following proposition

Proposition 3.10. *Let $G = (V, D)$ be a DAG with n nodes and $\mathcal{P}$ be an equal variance partition of G. Then, $\mathcal{M}^{\mathcal{P}}(G) \subset PD_n$.*

As before, we can parametrize the LSEM of a DAG G with

$$\mathcal{M}^{\mathcal{P}}(G) = \phi_G(\mathbb{R}^D, DP_n^{\mathcal{P}}). \qquad (22)$$

We define an extended version of covariance equivalence of LSEMs for arbitrary groups of equal error variances.

Definition 3.11. Let G, G' be DAGs with n nodes and $\mathcal{P}$ be an equal error variance partition of G.

(a) We say G and G' are *covariance equivalent under $\mathcal{P}$* if $\mathcal{M}^{\mathcal{P}}(G) = \mathcal{M}^{\mathcal{P}}(G')$.

(b) The *covariance equivalence class* of G under $\mathcal{P}$ is the set of all graphs G' such that G and G' are covariance equivalent under $\mathcal{P}$.

Now, we rephrase the results from the two previous sections with the notation just introduced.

Proposition 3.12. *Let G, G' be DAGs with n nodes and $\mathcal{P}$ be a variance partition of G.*

(i) *If $\mathcal{P} = \{\{i\} \mid i \in [n]\}$, then G and G' are covariance equivalent under $\mathcal{P}$ if and only if G and G' are Markov equivalent.*

(ii) *If $\mathcal{P} = \{[n]\}$, then G and G' are covariance equivalent under $\mathcal{P}$ if and only if $G = G'$.*

Proof. That identical global Markov properties imply identical LSEMs directly follows from Corollary 2.17. Conversely, assume that G and G' do not have identical global Markov properties. It turns out that in Theorem 2.14(b), which is the completeness of the global Markov property, we can always assume that the distribution is Gaussian [6, Thm. 3]. Hence, without loss of generality, there exists a Gaussian distribution with

covariance matrix Σ that satisfies the global Markov property of G but not the one of G'. Then, Corollary 2.17 yields that $\Sigma \in \mathcal{M}(G) \setminus \mathcal{M}(G')$. For statement (ii), let $\mathcal{P} = \{[n]\}$. It is clear from Definition 3.9 that identical variance partitions and identical graphs imply identical models. Conversely, assume $\mathcal{M}^{\mathcal{P}}(G) = \mathcal{M}^{\mathcal{P}}(G')$. For a DAG with n vertices and directed edge set D define

$$\mathbb{R}^D_{full} := \{\Lambda = (\lambda_{ij}) \in \mathbb{R}^{n \times n} \mid \lambda_{ij} = 0 \text{ if and only if } i \to j \notin D\}. \tag{23}$$

Denote $G = (V, D)$ and $G' = (V, D')$. Since $\mathbb{R}^D_{full} \neq \emptyset \neq DP^{\mathcal{P}}_n$ for any DAG G, we can choose $\Sigma \in \mathcal{M}^{\mathcal{P}}(G)$ with parametrization $\Sigma = \phi_G(\Lambda, \Omega)$ where $(\Lambda, \Omega) \in \mathbb{R}^D_{full} \times DP_n$. Since $\mathcal{M}^{\mathcal{P}}(G) = \mathcal{M}^{\mathcal{P}}(G')$, we can also parametrize $\Sigma = \phi_{G'}(\Lambda', \Omega')$ where $\Lambda' = (\lambda')_{ij} \in \mathbb{R}^{D'}$ and $\Omega' \in DP^{\mathcal{P}}_n$. Define $\bar{G} = (V, \bar{D})$ as the subgraph of G' that has all edges $i \to j$ with $\lambda'_{ij} = 0$ removed. Then, $\Lambda' \in \mathbb{R}^{\bar{D}}_{full}$ and clearly $\Sigma = \phi_{\bar{G}(\Lambda', \Omega') \in \mathcal{M}^{\mathcal{P}}(\bar{G})}$. Now Theorem 3.6 yields $G = \bar{G}$ and thus $D = \bar{D} \subset D'$. By switching G and G' we obtain that $D \supset D'$ by the same argument and thus $D = D'$. Since G and G' have identical nodes, we get $G = G'$. $\qquad \square$

At the time of this writing, there is no known result on covariance equivalence of the form Proposition 3.12 for an arbitrary variance partition $\mathcal{P}$. This is the leading question of this thesis that we will explore in the next section.

Remark 3.13. LSEMs are constrained versions of the more general structural equation models, cf. the paper from Peters and Buehlmann [20, Section 1]. The question of identifying the underlying graph from a LSEM under groupwise equal error variances can thus be interpreted as identifying the underlying graph of a structural equation model under the additional assumptions of linear functions and Gaussian additive error terms with groupwise equal error variances. We remark that it has been shown that two other sets of assumptions can uniquely identify the underlying graph of a structural equation model referenced by Peters and Bühlmann [20]: linear functions and non-Gaussian noise [22] as well as additive noise components under exclusion of some function-noise combinations [10].

4   Computational exploration of arbitrary groups of equal error variances

Instead of working with the LSEM under some equal variance partition, we will go over to work with the ideal of polynomials vanishing on the model and utilize computational algebra to check covariance equivalence through ideal equality. We will give a description of an algorithm to efficiently compute all covariance equivalence classes under arbitrary equal error variance partitions implemented in the algebraic geometry software Macaulay2 with optional Maple usage. Finally, the new material of this thesis will be presented; the results of the algorithm on three and four nodes, i.e. the covariance equivalence classes under all equal error variance partitions on three and four nodes. Further interpretation of these results and additional use cases of the algorithm will be given in section 5.

4.1   Algebraic tools for computing covariance equivalence

This section mainly follows the idea of Drton [3, Chapter 9] applied to the case of group-wise equal error variances with some additional background material about computational algebra from Sullivant [24, Chapter 3.3] and Cox, Little, O'Shea [2, Chapter 2]. We denote the vanishing ideal by $\mathcal{I}(\cdot)$, the ideal generated by some polynomials by $\langle \cdot \rangle$, and the algebraic variety of an ideal by $V(\cdot)$. Let $G := (V, D)$ be a DAG with n vertices and $\mathcal{P}$ be a variance partition of G. Instead of looking at the model itself, we look at all polynomials that vanish on the entries of all $\Sigma \in \mathcal{M}^{\mathcal{P}}(G) \subset \mathbb{R}^{n \times n}$, i.e. the vanishing ideal

$$\mathcal{I}^{\mathcal{P}}(G) := \mathcal{I}(\mathcal{M}^{\mathcal{P}}(G)) = \{f \in \mathbb{R}[\Sigma] : f(\Sigma) = 0 \; \forall \Sigma \in \mathcal{M}^{\mathcal{P}}(G)\}. \tag{24}$$

From Equation (24) it is clear that identical models imply identical vanishing ideals. The converse, i.e. that identical ideals imply identical models, only follows from Equation (24) if $\mathcal{M}^{\mathcal{P}}(G)$ is an algebraic variety. This is unfortunately never the case, see Remark 4.9. It turns out, that identical ideals still imply identical models, but we need a more advanced idea for a proof.

Theorem 4.1. *Let G be a DAG with n nodes and $\mathcal{P}$ be a partition of G. Then holds*

$$\mathcal{M}^{\mathcal{P}}(G) = V(\mathcal{I}^{\mathcal{P}}(G)) \cap PD_n. \tag{25}$$

Sketch of proof. In the case of arbitrary error variances, the equality $\mathcal{M}(G) = V(\mathcal{I}(G)) \cap PD_n$ follows from Richardson and Sprites [21, Thm. 8.1]. In the review from Drton [3, Eq. (10.1)], a formulation close to the notation of this thesis can be found. Since the parametrization ϕ_G has a rational inverse by Drton [3, Thm 7.1], each error variance ω_{ii} is a rational function of the entries of $\Sigma \in \mathcal{M}^{\mathcal{P}}(G)$. Thus, any equal error variance constraint of the type $\omega_{ii} = \omega_{jj}$ can be translated into some equation $a/b = c/d$ where a, b, c, d are polynomials on the entries of Σ. Since the inverse of ϕ_G always exists and b, d are denominators, the polynomials b, d cannot be zero so we can multiply the equation with b, d to obtain the polynomial equation $ad = bc$. Hence, all equal error variance constraints can be written as polynomial constraints on the entries of Σ. Let $p_1, \ldots, p_k$ be all these polynomial constraints. Then we have $\mathcal{I}^{\mathcal{P}}(G) = \mathcal{I}(G) + \langle p_1, \ldots, p_k \rangle$ and

$\mathcal{M}^{\mathcal{P}}(G) = \mathcal{M}(G) \cap V(\langle p_1, \ldots, p_k \rangle)$. Using the case with no equal variance assumptions and basic properties of ideals and varieties we have that

$$\mathcal{M}^{\mathcal{P}}(G) = \mathcal{M}(G) \cap V(\langle p_1, \ldots, p_k \rangle) = V(\mathcal{I}(G)) \cap V(\langle p_1, \ldots, p_k \rangle) \cap PD_n =$$
$$V(\mathcal{I}(G) + \langle p_1, \ldots, p_k \rangle) \cap PD_n = V(\mathcal{I}^{\mathcal{P}}(G)) \cap PD_n. \quad (26)$$

$\square$

Corollary 4.2. *Let G, G' be DAGs with n vertices and $\mathcal{P}$ be partition of G. Then, $\mathcal{M}^{\mathcal{P}}(G) = \mathcal{M}^{\mathcal{P}}(G')$ if and only if $\mathcal{I}^{\mathcal{P}}(G) = \mathcal{I}^{\mathcal{P}}(G')$.*

Proof. If the models are equal, then the ideals are equal by definition, cf. Equation (24). The converse statement follows from Theorem 4.1. $\square$

From Corollary 4.2 follows that DAGs are covariance equivalent under any equal variance partition if and only if their vanishing ideals are identical. This opens the door to using computational algebra to verify covariance equivalence. The remainder of this section outlines an algorithm for computationally verifying covariance equivalence without assuming any prior knowledge on computational algebra.

To check ideal equality computationally, we need Gröbner basis. We will give a very brief overview of their definition and properties that are useful in our setting and forward interested readers to Cox, Little, and O'Shea [2, Chapter 2]. Gröbner basis are an important tool in computational algebra that allow to store and perform calculations with ideals on computers conveniently. They are motivated by the fact that many questions around ideals can easily be answered if only considering monomial ideals [2, Chapter 2.4], i.e. ideals that are generated by only monomials. To apply results from these simpler cases, an order on all monomials with some properties is defined and arbitrary polynomials are associated with their highest ordered monomial. In the following, we consider $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ and the polynomial ring $\mathbb{K}[p] := \mathbb{K}[p_1, ..., p_r]$ with $r \in \mathbb{N}$.

Definition 4.3. A *term ordering* $\prec$ is a total ordering on the set of all monomials of $\mathbb{K}[p]$ such that

(a) $1 = p^0 \preceq p^u$ for all $u \in \mathbb{N}^r$ and

(b) $p^u \prec p^v$ implies $p^w \cdot p^u \prec p^w \cdot p^v$ for all $w \in \mathbb{N}^r$.

Definition 4.4. The greatest monomial of a polynomial $f \in \mathbb{K}[p]$ with respect to the term ordering $\prec$ is called *initial monomial* and denoted by $\text{in}_{\prec}(f)$.

We emphasize that $\prec$ is a total ordering by definition, thus all monomials can be compared and the initial monomial is well-defined. A simple example of a term ordering is the lexicographical term ordering $\prec_{lex}$, which has $p^u \prec_{lex} p^v$ if and only if the first non-zero entry from the left of $u - v$ if negative. Now, let us define Gröbner basis.

Definition 4.5. Let $I \subset \mathbb{K}[p]$ be an ideal and $\prec$ be a term ordering.

(a) The *initial ideal* of I is defined as $\text{in}_{\prec}(I) := \langle \text{in}_{\prec}(f) \mid f \in I \rangle$.

(b) A *Gröbner basis* of the ideal I with respect to the term ordering $\prec$ is a finite subset $G \subset I$ such that the initial ideals of I and G are identical, i.e.

$$\langle \mathrm{in}_{\prec}(g) : g \in G \rangle = \langle \mathrm{in}_{\prec}(f) : f \in I \rangle. \tag{27}$$

We note that Gröbner basis of an ideal with respect to different term orderings can be different. The following example from Sullivant [24, p. 49] shows that $\mathrm{in}_{\prec}(\langle f_1, ..., f_m \rangle)$ and $\langle \mathrm{in}_{\prec}(f_1), ..., \mathrm{in}_{\prec}(f_m) \rangle$ are not equal in general.

Example 4.6. Let $r = 2$ and $I = \langle p_1^2, p_1 p_2 + p_2^2 \rangle$ with lexicographic term ordering $\prec := \prec_{lex}$. Denote the ideal of the initial monomials of the generators of I by $J = \langle \mathrm{in}_{\prec}(p_1^2), \mathrm{in}_{\prec}(p_1 p_2 + p_2^2) \rangle = \langle p_1^2, p_1 p_2 \rangle$. It is clear that $p_2^3 \notin J$ but $p_2 p_1^2 + (p_2 - p_1)(p_1 p_2 + p_2^2) = p_2^3 \in I$.

A Gröbner basis G of the ideal I with respect to a term ordering $\prec$ fulfills similar properties to a basis of finite dimensional vector spaces; G is a finite subset of I by definition, G generates I, i.e. $\langle G \rangle = I$ [24, Cor. 3.3.10], and the Hilbert basis theorem yields that a Gröbner basis exists for every ideal and every term ordering [24, p. 50]. Although Gröbner basis are not unique for a fixed term ordering, we can define *reduced* Gröbner basis by norming all coefficients of the leading monomials and removing all redundant polynomials from a given Gröbner basis [2, Def. 2.7.4]. Given any ideal and any term ordering, it turns out that such a reduced Gröbner basis always exists and that this basis is unique [2, Thm. 2.7.5]. This yields a convenient way to compare ideals by fixing a term ordering and comparing their reduced Gröbner basis with respect to that term ordering. To make use of Gröbner basis in practice, *Buchberger's algorithm* [2, Chapter 2.7] provides a constructive way to compute a Groebner basis for any ideal $I = \langle f_1, \ldots, f_k \rangle$ with respect to a term ordering $\prec$. The algorithm exploits *Buchberger's criterion* [2, Thm. 2.6.6] that gives a sufficient and necessary condition for a finite $G \subset I$ being a Gröbner basis of I with respect to $\prec$.

Example 4.7. With Buchberger's criterion, it can be shown that $G = \{p_1^2, p_1 p_2 + p_2^2, p_2^3\}$ is a Gröbner basis of ideal $I = \langle p_1^2, p_1 p_2 + p_2^2 \rangle$ with respect to the lexicographic term ordering [24, Example 3.3.13], see Example 4.6.

In each step, the Buchberger's algorithm checks if G fulfills Buchberger's criterion and if not, adds a specific polynomial computed from G. We remark that Buchberger's algorithm always terminates but has hardly predictable runtimes that can be doubly-exponential in the number of variables in the worst-case [18]. Moreover, the choice of the term ordering has a profound influence on the performance of the algorithm. Usually, computer algebra systems perform Buchberger's algorithm in a way such that the resulting Gröbner basis can easily be reduced computationally [2, p. 94]. In summary, we can now computationally verify ideal equality based on reduced Gröbner basis given any finite generating sets of the ideals to compare. It remains to compute such a finite generating set for the vanishing ideal of our model, cf. Equation (24).

For a graph G with n vertices and a variance partition $\mathcal{P}$ of G we consider the polynomial parametrization of the model $\mathcal{M}^{\mathcal{P}}(G)$, i.e.

$$\mathcal{I}^{\mathcal{P}}(G) = \mathcal{I}(\mathcal{M}^{\mathcal{P}}(G)) = \mathcal{I}(\phi_G(\mathbb{R}^D, DP_n^{\mathcal{P}})). \tag{28}$$

The general problem of describing the vanishing ideal of the image of an algebraic variety under a polynomial map is called *implicitization problem* and has a well-known solution. It turns out that we can turn the set in the domain of ϕ_G in Equation (28) into a variety without changing the vanishing ideal by including any diagonal matrices that satisfy the equal variance constraints instead of just the ones with positive entries. Define this set as

$$D_n^{\mathcal{P}} := \{\operatorname{diag}(a_1, ..., a_n) \in \mathbb{R}^{n \times n} \mid \forall i, j \in [n] : (\exists P \in \mathcal{P} : \{i, j\} \subset P \Rightarrow a_i = a_j)\}. \quad (29)$$

Proposition 4.8. *Let $G = (V, D)$ be a DAG with n vertices and let $\mathcal{P}$ be a variance partition of G. Define $\mathcal{C}_G^{\mathcal{P}} := \phi_G^{\mathcal{P}}(\mathbb{R}^D, D_n^{\mathcal{P}})$. Then holds*

$$\mathcal{I}^{\mathcal{P}}(G) = \mathcal{I}(\mathcal{C}_G^{\mathcal{P}}). \quad (30)$$

Proof. We clearly have $\mathcal{M}^{\mathcal{P}}(G) = \phi_G(\mathbb{R}^D, DP_n^{\mathcal{P}}) \subset \mathcal{C}_G^{\mathcal{P}}$ so $\mathcal{I}^{\mathcal{P}}(G) \supset \mathcal{I}(\mathcal{C}_G^{\mathcal{P}})$. Conversely, let $f \in \mathcal{I}^{\mathcal{P}}(G)$ and $\Lambda \in \mathbb{R}^D$. We write $\mathcal{P} = \{P_1, ..., P_p\}$ for some $p \leq n$ and define

$$M_k := \{\operatorname{diag}(m_1, ..., m_n) \in \mathbb{R}^{n \times n} \mid \forall j > k \; \forall i \in P_j : m_i > 0\} \quad (31)$$

for $k \in [p] \cup \{0\}$. The diagonal matrices in M_k can be described by having positive values for the entries of some sets in the partition and any values for the entries of the remaining sets of the partition. We claim that for any $k \in [p] \cup \{0\}$, the polynomial f vanishes on all $\phi_G(\Lambda, \Omega)$ with $\Omega \in M_k$. We show this by induction over k. Since $M_0 = DP_n^{\mathcal{P}}$ and $\mathcal{M}^{\mathcal{P}}(G) = \phi_G(\mathbb{R}^D, DP_n^{\mathcal{P}})$ it is clear that the claim holds for M_0. Now, assume the claim holds for M_k. Without loss of generality we denote $P_{k+1} = \{1, ..., r-1\}$ for some $r \leq n$. Let $\Omega = (\omega_{ij}) \in M_k$ and define $\Omega_x := \operatorname{diag}(x, ..., x, \omega_{rr}, ..., \omega_{nn})$ for $x \in \mathbb{R}$. Then, $\Omega_x \in M_k$ for $x > 0$. Since ϕ_G and f are polynomial maps, we can define the polynomial

$$g(x) := f(\phi_G(\Lambda, \Omega_x)) \in \mathbb{R}[x] \quad (32)$$

Then, $g(x) = 0$ for all $x > 0$ by induction assumption. Since $\mathbb{R}$ is an infinite field and g is polynomial in one variable with the infinite roots $x > 0$, it follows that $g = 0$ so in particular $g(x) = 0$ for all $x \in \mathbb{R}$. Since $\Omega \in \mathcal{M}_k$ arbitrary, we obtain $f(\phi_G(\Lambda, \Omega)) = 0$ for all $\Omega \in M_{k+1}$. This completes the induction. Hence we have shown that f vanishes on $\phi_G(\Lambda, \Omega)$ for any $\Omega \in D_n^{\mathcal{P}} = M_p$. Since $\Lambda \in \mathbb{R}^D$ arbitrary, f vanishes on all elements of $\phi_G(\mathbb{R}^D, DP_n) = \mathcal{C}_G$. $\qquad \square$

Remark 4.9. Proposition 4.8 implies that $\mathcal{M}^{\mathcal{P}}(G)$ is never an algebraic variety. For any DAG G and variance partition $\mathcal{P}$, we have $0 \in D_n^{\mathcal{P}}$ so $0 \in \mathcal{C}_G \subset V(\mathcal{I}(\mathcal{C}_G)) = V(\mathcal{I}(\mathcal{M}^{\mathcal{P}}(G)))$ but $0 \notin \mathcal{M}^{\mathcal{P}}(G) \subset PD_n$. Thus, $\mathcal{M}^{\mathcal{P}}(G) \subsetneq V(\mathcal{I}(\mathcal{M}^{\mathcal{P}}(G)))$.

Proposition 4.8 shows that $\mathcal{I}^{\mathcal{P}}(G) = \mathcal{I}(\phi_G(\mathbb{R}^D, D_n^{\mathcal{P}}))$ where $\mathbb{R}^D \times D_n^{\mathcal{P}}$ is an algebraic variety, so a solution of the implicitization problem would be useful for us.

Theorem 4.10 (Implicitization). *[24, Prop. 3.4.7] Define a polynomial ring over a field $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ with indeterminates $p_1, ..., p_n$ and $q_1, ..., q_m$ where $n, m \in \mathbb{N}$. Denote $\mathbb{K}[p] := \mathbb{K}[p_1, ..., p_n]$ and $\mathbb{K}[q] := \mathbb{K}[q_1, ..., q_m]$. Let $\phi : \mathbb{K}^m \to \mathbb{K}^n$ be a polynomial map and $V \subset \mathbb{K}^m$ an algebraic variety. Denote the graph of ϕ by*

$$\Gamma_\phi = \{(\phi(x), x) : x \in \mathbb{K}^m\} \subset \mathbb{K}^{n+m}. \quad (33)$$

Then the vanishing ideal of the image of ϕ under V is

$$\mathcal{I}(\phi(V)) = (\mathcal{I}(V) + \mathcal{I}(\Gamma_\phi)) \cap \mathbb{K}[p]. \quad (34)$$

Before we give a proof, let us have a closer look at Equation (34). We have $\mathcal{I}(V) \subset \mathbb{K}[q]$ and $\mathcal{I}(\Gamma_\phi) \subset \mathbb{K}[p, q]$ where $\mathbb{K}[p, q] := \mathbb{K}[p_1, ..., p_n, q_1, ..., q_m]$. Since $\mathbb{K}[q] \subset \mathbb{K}[p, q]$, we can properly add the ideals together and obtain an ideal in $\mathbb{K}[p, q]$. The intersection with $\mathbb{K}[p]$ removes any polynomial $f \in \mathbb{K}[p, q]$ that contains any $q_1, ..., q_m$ indeterminate which yields an ideal in $\mathbb{K}[p]$. Ideals like Equation (34) that intersect with a polynomial ring to eliminate some terms are also called *elimination ideals*.

Proof of Theorem 4.10. We follow the proof of Sullivant [24, Prop. 3.4.7]. Define

$$\Gamma_\phi(V) := \{(y, x) \in \mathbb{K}^{n+m} \mid x \in V, \ y = \phi(x)\}. \tag{35}$$

We first claim that

$$\mathcal{I}(\phi(V)) = \mathcal{I}(\Gamma_\phi(V)) \cap \mathbb{K}[p]. \tag{36}$$

Let π_p be the projection $\pi_p : \mathbb{K}^{n+m} \to \mathbb{K}^n$, $(a_1, ..., a_n, b_1, ..., b_m) \mapsto (a_1, ..., a_n)$. Then clearly $\pi_p(\Gamma_\phi(V)) = \phi(V)$ so we want to show that

$$\mathcal{I}(\pi_p(\tilde{V})) = \mathcal{I}(\tilde{V}) \cap \mathbb{K}[p] \tag{37}$$

with $\tilde{V} = \Gamma_\phi(V)$. We have that $f \in \mathcal{I}(\pi_p(\tilde{V}))$ if and only if f vanishes on $\pi_p(\tilde{V})$ and $f \in \mathbb{K}[p]$. But this is the case if and only if f vanishes on $\tilde{V}$ and has indeterminates only in $\mathbb{K}[p]$, which is precisely that $f \in \mathcal{I}(\tilde{V}) \cap \mathbb{K}[p]$. We note that Equation (36) holds for any algebraic variety $\tilde{V} \subset \mathbb{K}^{n+m}$. We obtain Equation (37). It remains to show that

$$\mathcal{I}(\Gamma_\phi(V)) = \mathcal{I}(V) + \mathcal{I}(\Gamma_\phi). \tag{38}$$

Let $f \in \mathcal{I}(\Gamma_\phi(V))$. Define $M := \{p_1 - \phi_1(q), ..., p_n - \phi_n(q)\}$, then clearly $\langle M \rangle = \mathcal{I}(\Gamma_\phi)$. Now we choose a term ordering $\prec$ such that $p_i \succ q_j$ for all $i \in [n]$ and $j \in [m]$. Similar to one-dimensional polynomial division, it turns out that we can always write

$$f = \sum_{i=1}^{n} h_i(p_i - \phi_i(q)) + r \tag{39}$$

with $h_i \in \mathbb{K}[p, q]$ and $r = 0$ or no monomial of r being divisible by any leading monomial of M [2, Thm 2.3.3]. Thus, by the choice of term order, no monomial of r is divisible by any $p_1, ..., p_n$, so $r \in \mathbb{K}[q]$. Since $\Gamma_\phi \supset \Gamma_\phi(V)$, we have $\langle M \rangle = \mathcal{I}(\Gamma_\phi) \subset \mathcal{I}(\Gamma_\phi(V))$ and hence $\sum_{i=1}^{n} h_i(p_i - \phi_i(q)) \in \mathcal{I}(\Gamma_\phi(V))$. Together, we obtain

$$r = f - \sum_{i=1}^{n} h_i(p_i - \phi_i(q)) \in \mathcal{I}(\Gamma_\phi(V)) \cap \mathbb{K}[q]. \tag{40}$$

We utilize the idea of the projection from above again, but this time with $\pi_q : \mathbb{K}^{n+m} \to \mathbb{K}^m$, $(a_1, ..., a_n, b_1, ..., b_m) \mapsto (b_1, ..., b_m)$ to obtain

$$\mathcal{I}(\Gamma_\phi(V)) \cap \mathbb{K}[q] = \mathcal{I}(\pi_q(\Gamma_\phi(V))) = \mathcal{I}(V). \tag{41}$$

Hence, we have $r \in \mathcal{I}(V)$ and thus

$$f = r + \sum_{i=1}^{n} h_i(p_i - \phi_i(q)) \in \mathcal{I}(V) + \langle M \rangle = \mathcal{I}(V) + \mathcal{I}(\Gamma_\phi) \tag{42}$$

as claimed. Conversely, all ideals vanishing on V also vanish on the latter entries of $\Gamma_\phi(V)$, i.e. $\mathcal{I}(\Gamma_\phi(V)) \supset \mathcal{I}(V)$, and from $\Gamma_\phi(V) \subset \Gamma_\phi$ follows $\mathcal{I}(\Gamma_\phi(V)) \supset \mathcal{I}(\Gamma_\phi)$. We obtain $\mathcal{I}(\Gamma_\phi(V)) \supset \mathcal{I}(V) + \mathcal{I}(\Gamma_\phi)$ since ideals are closed under addition. This completes the proof.   □

Let us apply implicization to the vanishing ideal of the model. The p indeterminates are the upper diagonal entries of Σ, which is sufficient since Σ is always symmetric, and the q indeterminates are the entries of Λ and Ω that can be non-zero. The remaining entries of Λ and Ω are set to zero. It is clear that the domain of ϕ_G is $\mathbb{R}^{|D|+|V|}$ and that $\mathbb{R}^D \times D_n^{\mathcal{P}} \subset \mathbb{R}^{|D|+|V|}$ is an algebraic variety described by the polynomials that encode the equal error variances along the diagonal of Ω. Moreover, $\langle \Sigma - \phi_G(\Lambda, \Omega) \rangle$ is a basis of $\mathcal{I}(\phi_G)$, cf. the set M in the proof of the implicitization theorem. We obtain

$$\mathcal{I}^{\mathcal{P}}(G) = (\langle \omega_{ii} - \omega_{jj} \mid \exists P \in \mathcal{P} : \{i, j\} \subset P \rangle + \langle \Sigma - \phi_G(\Lambda, \Omega) \rangle) \cap \mathbb{R}[\Sigma]. \tag{43}$$

We can further simplify Equation (43). Since $\Sigma - \phi_G(\Lambda, \Omega)$ vanishes if and only if $\Omega - (I - \Lambda)^T \Sigma (I - \Lambda)$ vanishes, we have

$$\mathcal{I}^{\mathcal{P}}(G) = \left(\langle \omega_{ii} - \omega_{jj} \mid \exists P \in \mathcal{P} : \{i, j\} \subset P \rangle + \langle \Omega - (I - \Lambda)^T \Sigma (I - \Lambda) \rangle \right) \cap \mathbb{R}[\Sigma]. \tag{44}$$

Now we can easily eliminate Ω by substitution with $K := (I - \Lambda)^T \Sigma (I - \Lambda) = (k_{ij})$. For the equal error variance assumptions, we can just replace ω_{ii} and ω_{jj} by k_{ii} and k_{jj}. Hence, $\Omega - (I - \Lambda)^T \Sigma (I - \Lambda) = 0$ if and only if the equal variance assumptions on the diagonal entries of K vanish and $k_{ij} = 0$ for $i \neq j$, since Ω is a diagonal matrix. This yields

$$\mathcal{I}^{\mathcal{P}}(G) = (\langle k_{ii} - k_{jj} \mid \exists P \in \mathcal{P} : \{i, j\} \subset P \rangle + \langle k_{ij} \mid i \neq j \rangle) \cap \mathbb{R}[\Sigma]. \tag{45}$$

It is clear, that a generating set of the ideal in Equation (45) before elimination is

$$J_G^{\mathcal{P}} := \{k_{ii} - k_{jj} \mid \exists P \in \mathcal{P} : \{i, j\} \subset P\} \cup \{k_{ij} \mid i \neq j\} \tag{46}$$

which we can easily be generated on a computer. It remains to compute a generating set of the elimination ideal.

Theorem 4.11. *[24, Thm. 3.4.6] Let $I \in \mathbb{K}[p_1, ..., p_m, q_1, ..., q_n]$ be an ideal and let G be a Gröbner basis of I with respect to a term ordering $\prec$ such that $q_i \succ p_j$ for all $i \in [n]$ and $j \in [m]$. Define $\mathbb{K}[p] := \mathbb{K}[p_1, ..., p_m]$, then*

$$G \cap \mathbb{K}[p] \tag{47}$$

is a Gröbner basis of the elimination ideal $I \cap \mathbb{K}[p]$ with respect to $\prec$.

Proof. We follow the proof from Sullivant [24, Thm. 3.4.6]. We directly show the definition of a Gröbner basis. Let $f \in I \cap \mathbb{K}[p]$, then we have $\mathrm{in}_\prec(f) \in \mathrm{in}_\prec(G)$ since G is a Gröbner basis of I. Since $\mathrm{in}_\prec(G)$ is a monomial ideal and $\mathrm{in}_\prec(f)$ is a monomial, this is equivalent to saying that there exists a $g \in G$ such that $\mathrm{in}_\prec(g)$ divides $\mathrm{in}_\prec(f)$ [2, Lem. 2.4.3]. Since $f \in \mathbb{K}[p]$, we have $\mathrm{in}_\prec(f) \in \mathbb{K}[p]$ and thus also $\mathrm{in}_\prec(g) \in \mathbb{K}[p]$. By our choice of term ordering, $\mathrm{in}_\prec(g) \in \mathbb{K}[p]$ implies that $g \in \mathbb{K}[p]$. Hence, $g \in G \cap \mathbb{K}[p]$ and by same argument as above, $\mathrm{in}_\prec(f) \in \mathrm{in}_\prec(G \cap \mathbb{K}[p])$. Since f arbitrary, $G \cap \mathbb{K}[p]$ is Gröbner basis of $I \cap \mathbb{K}[p]$.   □

A term ordering with the above property is also called *elimination ordering*. We remark that the term ordering we used in the proof of the implicitization theorem is not an elimination ordering that eliminates $q_1, ..., q_n$.

In summary, we can now compute whether two DAGs G and G' are covariance equivalent under some equal variance partition $\mathcal{P}$ with the following steps: compute the polynomials in $J_G^{\mathcal{P}}$, compute a Gröbner basis of the ideal $\langle J_G^{\mathcal{P}} \rangle$ with respect to an elimination ordering that eliminates the entries of Λ and Ω, compute the Gröbner basis of $\mathcal{I}^{\mathcal{P}}(G)$ with Theorem 4.11, and reduce that Gröbner basis . If we denote this reduced Gröbner basis by $R_G^{\mathcal{P}}$ and do the same to obtain $R_{G'}^{\mathcal{P}}$, the two DAGs G and G' are covariance equivalent under $\mathcal{P}$ if and only if $R_G^{\mathcal{P}} = R_{G'}^{\mathcal{P}}$. Luckily, large parts of these steps are already implemented in computer algebra systems as we will see in the next section.

4.2   Implementation in Macaulay2 with Maple add-on

In the previous section, we have outlined a way to compute and compare the vanishing ideals of linear structural equation models of any DAG under any equal variance partition using computational algebra. Let us see how these results can be implemented for a simple example in the free-to-use algebraic geometry software Macaulay2 [8].

Example 4.12. The following Macaulay2 code snippet shows how to compute the vanishing ideal of $G_1 = 1 \to 2 \to 3$ from Example 2.12 under $\mathcal{P} = \{\{1, 2\}, \{3\}\}$. The variable `vanishIdeal` corresponds to $\mathcal{I}^{\mathcal{P}}(G_1)$ and evaluates to an ideal with generators `s13s22 - s12s23, s11^2 + s12^2 - s11s22`. Note that Macaulay2 starts indexing at 0. To compute covariance equivalence of two DAGs G, G', we just execute the code below twice for the corresponding `Lambda` matrices and compare the resulting ideals with `vanishIdeal == vanishIdeal'`. The command `==` is a built-in command that computes and compares the reduced Gröbner basis of the ideals internally.

```
-- create the polynomial ring
R = QQ[l12,l13,l21,l23,l31,l32,s11,s12,s13,s22,s23,s33]

-- compute matrices from the model
Sigma = matrix{{s11,s12,s13},{s12,s22,s23},{s13,s23,s33}}
Lambda = matrix{{0,l12,0},{0,0,l23},{0,0,0}}
K = transpose(id_(R^3) - Lambda) * Sigma * (id_(R^3) - Lambda)

-- compute vanishing ideal
I = ideal({K_(0,1),K_(0,2),K_(1,2), K_(0,0) - K_(1,1)})
vanishIdeal = eliminate({l12,l13,l21,l23,l31,l32},I)
```

Now, we want to scale this idea to compute all covariance equivalence classes for n nodes under any equal variance partitions. We observe that the structure of a graph and its equal variance partition do not change by permuting the labels. Thus, it seems intuitive that computing the covariance equivalence classes of both variance partitions $\{\{1, 2\}, \{3\}\}$ and $\{\{1, 3\}, \{2\}\}$ does not provide any new information since both partitions have two nodes with equal error variances and one node with arbitrary error variance. Let us make

this idea rigorous. The labels of a DAG correspond to the positions in the matrices Λ, Ω, Σ of the model, e.g. λ_{ij} corresponds to the edge from label i to label j. Hence, the following holds.

Proposition 4.13. *Let G be a DAG with n nodes, let $\mathcal{P}$ be an equal variance partition of G, and let $\sigma \in S_n$ be a permutation of $[n]$. Then, the vanishing ideal of G under $\mathcal{P}$ with labels permuted by σ is simply $\mathcal{I}^{\mathcal{P}}(G)$ with indices of the indeterminates permuted by σ.*

Example 4.14. Using Proposition 4.13 and the Macaulay2 notation and result from Example 4.12, the generators of the vanishing ideal of $G_3 = 3 \rightarrow 2 \rightarrow 1$ under $\mathcal{P} = \{\{3, 2\}, \{1\}\}$ are `s13s22 - s23s12, s33^2 + s23^2 - s33s22`. We note that G_3 already appeared in Example 3.1. Moreover, we remark that we only defined the upper half of the symmetric covariance matrix Σ of the model as indeterminates, so we need to be careful when permuting the indices of the 's' indeterminates.

We say two variance partitions $\mathcal{P}$ and $\mathcal{P}'$ of $[n]$ are *isomorphic*, if $\mathcal{P}$ can be obtained by permuting the labels of $\mathcal{P}'$. It is clear that this relation is an equivalence relation on the partitions of $[n]$. Let $\mathcal{P}$ be any partition of $[n]$, let $\sigma \in S_n$ be a permutation of $[n]$, and define $\mathcal{P}' = \sigma(\mathcal{P})$ isomorphic to $\mathcal{P}$, where $\sigma(\cdot)$ denotes a permutation of the labels by σ. By Proposition 4.13 and Corollary 4.2, the DAGs G and G' are covariance equivalent under $\mathcal{P}$ if and only if $\sigma(G)$ and $\sigma(G')$ are covariance equivalent under $\sigma(\mathcal{P}) = \mathcal{P}'$. Hence, computing covariance equivalence classes of only one representative of each equivalence class of partitions of $[n]$ is sufficient. Let B_n be the set of representative partitions such that we can order the sets in each partition in B_n by descending cardinality and read the labels in ascending order from left to right skipping all curly braces. We call B_n *base partitions* of n and will only consider these base partitions in the remainder of the thesis.

Example 4.15. We have $B_3 = \{\{\{1\}, \{2\}, \{3\}\}, \{\{1, 2\}, \{3\}\}, \{[3]\}\}$.

A straight-forward algorithm to compute these sufficient covariance equivalence classes is described in Macaulay2 pseudocode by Algorithm 1. Let us comment on some implicit functions in the pseudocode: `vanishingIdeal` computes the steps outlined in Example 4.12, `compareVanIdeals` computes the equivalence classes based on vanishing ideal equality, and `generateAllDags` computes all DAGs with n nodes by generating all simple directed graphs with n nodes and removing the cyclic ones. Although the latter implementation is inefficient and could be improved significantly, it will be sufficient for our purposes.

To improve the performance of Algorithm 1, we further exploit Proposition 4.13. Let D_n denote the set of all DAGs with n nodes, then we only want to compute the vanishing ideals of a minimal subset $A_n \subset D_n \times B_n$ and obtain the remaining vanishing ideals by permutation. Similarly as before, we say two DAG partition tuples $(G, \mathcal{P})$ and $(G', \mathcal{P}')$ are *isomorphic* if $(G, \mathcal{P})$ can be obtained by permuting the labels of $(G', \mathcal{P}')$. Again, it is clear that this relation is an equivalence relation on $D_n \times B_n$ and that A_n can be any set that contains exactly one representative per equivalence class of n nodes. Giving a constructive combinatorical description of any A_n is beyond the scope of this thesis and also not necessary for our purposes. After computing a vanishing ideal under $\mathcal{P}$, we simply permute the labels by all possible permutations and save the permuted ideals with permuted partition equal to $\mathcal{P}$ that were not known before. We note that we can treat

---

Algorithm 1: Base version

    input : Number of nodes n

    output: One file per base variance partition containing the covariance
                equivalence classes

1 $allDags \leftarrow$ `generateAllDags`(n) ;

2 $baseVarPart \leftarrow$ `generateBasePartitions`(n);

3   **for** $\mathcal{P} \in baseVarPart$ **do**

4 $vanIdealDict \leftarrow$ **new** `MutableHashMap`;

5   **for** $G \in allDags$ **do**

6 $vanIdealDict\#G \leftarrow$ `vanishingIdeal`$(G, \mathcal{P})$;

7 $covEqClasses \leftarrow$ `compareVanIdeals`$(vanIdealDict)$;

8   `saveResultsToFile`$(\mathcal{P}, covEqClasses)$;

---

the base partitions separately in this step as they are not isomorphic to one another and thus cannot interfere when permuting the vanishing ideals. Since computing the vanishing ideal is a very expensive operation as it involves the computation of a Gröbner basis, this speeds up the algorithm substantially. The Macaulay2 pseudocode is given in Algorithm 2. To give an idea of the magnitude of improvement, we compare the cardinalities of

---

Algorithm 2: Improved version

    input : Number of nodes n

    output: One file per base variance partition containing the covariance
                equivalence classes

1 $allDags \leftarrow$ `generateAllDags`(n);

2 $baseVarPart \leftarrow$ `generateBasePartitions`(n);

3 $allPermus \leftarrow$ `generateAllPermutations`(n);

4   **for** $\mathcal{P} \in baseVarPart$ **do**

5 $vanIdealDict \leftarrow$ **new** `MutableHashMap`;

6 $validPermus \leftarrow \{\sigma \in allPermus \mid \sigma(\mathcal{P}) = \mathcal{P}\}$;

7   **for** $G \in allDags$ **do**

8   **if** $G \in$ `keys`$(vanIdealDict)$ **then**

9   `continue` ;

10 $I \leftarrow$ `vanishingIdeal`$(I, \mathcal{P})$;

11 $vanIdealDict\#G \leftarrow I$;

12   **for** $\sigma \in validPermus$ **do**

13   **if** $\sigma(G) \notin$ `keys`$(vanIdealDict)$ **then**

14 $vanIdealDict\#\sigma(G) \leftarrow \sigma(I)$;

15 $covEqClasses \leftarrow$ `compareVanIdeals`$(vanIdealDict)$;

16   `saveResultsToFile`$(\mathcal{P}, covEqClasses)$;

A_n and $D_n \times B_n$. We can explicitly compute the cardinality of $D_n \times B_n$ by multiplying the number of labeled DAGs with n nodes [12] for $|D_n|$ and the number of partitions of

the integer n [11] for $|B_n|$. For $|A_n|$, we obtain values from executing the algorithm for $n \in \{3, 4, 5\}$. The results can be seen in Table 1. We see a substantial improvement using $|A_n|$ instead of $D_n \times B_n$. We note that we could further reduce the runtime by running the

Nodes n	Base version $|D_n| \cdot |B_n|$	Improved version $|A_n|$	Quotient $|D_n||B_n|/|A_n|$
3	$25 \cdot 3 = 75$	26	2.89
4	$543 \cdot 5 = 2172$	393	5.52
5	$29281 \cdot 7 = 175686$	13714	12.81

Table 1: Number of vanishing ideals needed to compute for the two different versions of the algorithm.

for-loops in lines 4 and 7 of Algorithm 2 in parallel or improving the generation of DAGs through constructive procedures or parallelisation. However, computing the elimination ideal turns out to be the bottleneck as we will see in the next section.

Finally, let us turn to more technical aspects. All code is written and tested in Macaulay2 under version 1.19.1 [8]. In Macaulay2, we use the built-in command `eliminate` to calculate the elimination ideals, and the `==` operator to check for ideal equality. Since the Macaulay2 elimination command turned out to be slow in practice, we offer the possibility to outsource ideal elimination and comparison to Maple 2022 [16] via the Macaulay2 `run` command. In Algorithm 2, these are the functions `vanishingIdeal` and `compareVanIdeals`. In Maple, we use the `EliminationIdeal` command and twice the `IdealContainment` command of the package `PolynomialIdeals` to calculate the elimination ideal and check for ideal equality. All code with documentation is available on Github at:

https://github.com/LeWaldm/covarianceEquivalence

4.3 Results on three and four nodes

Now, we want to investigate the results of the algorithm from the previous section. Let n denote the number of nodes. We remark that the case $n = 2$ is completely covered by the known results from Section 3 so only $n > 2$ is considered. All code was executed on a machine with 16GB memory and two physical cores at 2.50 GHz (Intel® Core™ i5-4200M). This machine could only compute the results for $n \in \{3, 4\}$ and crashed for $n = 5$ due to insufficient memory while computing a specific elimination ideal in Maple. To obtain results for $n \geq 5$, more memory is needed before applying any other improvements mentioned in the previous section. The runtimes of the two versions of the algorithm from Section 4.2 for $n \in \{3, 4\}$ nodes with and without utilizing Maple to perform elimination and ideal comparison can be seen in Table 2. All versions of the algorithm worked very well for $n = 3$, that the versions with Maple were slower is due to computational overhead needed to communicate between Macaulay2 and Maple. For $n = 4$, the versions with Maple behaved substantially better. In particular, the calculation of a specific elimination ideal in Macaulay2 was interrupted after 72h without termination. The computation of this particular elimination ideal took less than a minute in Maple. It is well known that Gröbner basis computations can be improved significantly by prioritizing the elements

nodes	Base version	Base version with Maple	Improved version with Maple
3	2.5s	8s	3.3s
4	>72h	70min	21min

Table 2: Runtimes of the base and the improved version with and without the aid of Maple for three and four nodes.

that are added in the Buchberger's algorithm [18], see, e.g., the M4GB algorithm from Makarim and Stevens [15]. However, the tremendous performance difference between Macaulay2 and Maple came as a surprise. With Maple, we see that the improved version ran 3.3 times faster than the base version. That the expected speed up of around 5.52 from Table 1 was not observed is likely due to the different sizes of the permutation equivalence classes of DAGs and the high variation in the time needed to compute vanishing ideals. Now, let us have a look at the actual results, i.e., the covariance equivalence classes of the partitions not dealt with in Section 3.1 and Section 3.2. Table 3 counts the number of classes by their number of members. Readers interested in similar properties of Markov equivalence classes up to ten nodes may consult the results from Gillispie and Perlmann [7]. We see that all partitions but $\mathcal{P} = \{\{1, 2\}, \{3\}, \{4\}\}$ result in unique models. Let us

n	$|D_n|$	Partition	Class size distribution
3	25	$\{\{1,2\},\{3\}\}$	with 1 member: 25
4	543	$\{\{1,2\},\{3\},\{4\}\}$	with 1 member: 405 with 2 members: 69
		$\{\{1,2\},\{3,4\}\}$	with 1 member: 543
		$\{\{1,2,3\},\{4\}\}$	with 1 member: 543

Table 3: Distribution of covariance equivalence class sizes under the base partitions $|B_n|$ not covered by the cases discussed in Section 3 for three and four nodes.

focus on $\mathcal{P}$ for the remainder of this section. All covariance equivalence classes under $\mathcal{P}$ with more than one member contain precisely two graphs that only change the direction of the edge between node 3 and 4. If a graph G is in such a covariance equivalence class, we say that the edge between 3 and 4 in G is *invertible*. The edge between node 3 and 4 is not always invertible. Figure 2 depicts some graphs (a),(b),(c), and (d) where the edges between 3 and 4 of (a) and (b) but not the one of (c) are invertible.

Remark 4.16. In general, we cannot deduce the covariance equivalence class of a graph with isolated nodes from lower dimensional cases disregarding the isolated nodes in the graph and the partition. Take graph (d) in Figure 2, denote the subgraph of (d) that removes node 1 with G' and denote $\mathcal{P}' = \{\{2\}, \{3\}, \{4\}\}$ that removes node 1 from $\mathcal{P}$. Then, the covariance equivalence class of G' under $\mathcal{P}'$ is the Markov equivalence graph of G' and contains, e.g., $G'' = 4 \to 2 \to 3$. However, the graph with node 1 added again as isolated node to G'' is not covariance equivalent to (d) under $\mathcal{P}$ since (d) is in a covariance equivalence class under $\mathcal{P}$ on its own. This makes sense as assuming that node 1 and 2 have equal error variance adds another polynomial to the vanishing ideal although node 1 is isolated, cf. Equation (45).
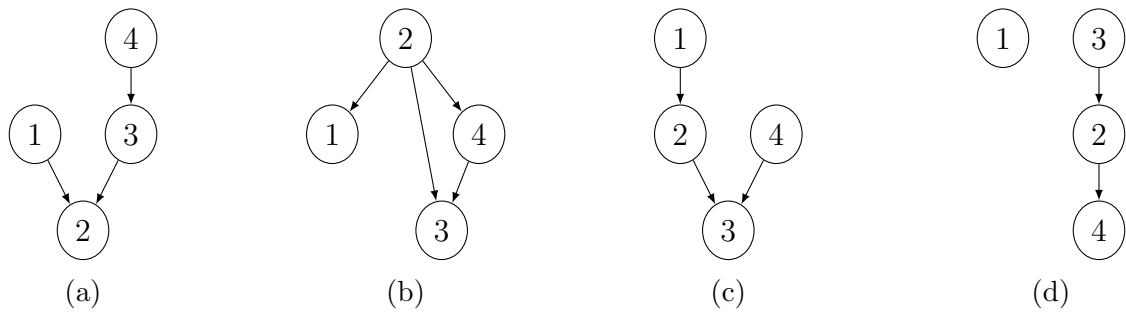
Figure 2: Some DAGs with four nodes under $\{\{1,2\},\{3\},\{4\}\}$ where graph (a) and (b) are in covariance equivalence classes with two members and graph (c) and (d) are in covariance equivalence classes on their own.

All graphs where the edge $3 \rightarrow 4$ is invertible can be seen in the appendix. In the next section, we will conjecture a precise characterisation of covariance equivalence based on the results of this section.

5   Interpretation of results and extension to cyclic directed graphs

Based on the results from Section 4.3, we will state a conjecture on covariance equivalence of DAGs with arbitrary number of nodes under any variance partition. This conjecture entails the two known equal error variance settings from Section 3 and explains the computational results on three and four nodes. Assuming that the conjecture holds true, we will compute the distribution of covariance equivalence class sizes up to six nodes and give a conjecture on their possible sizes under arbitrary number of nodes and arbitrary variance partitions. Finally, we will see how we can apply adjustments to Section 2.1 and Section 4.2 to compute and compare the vanishing ideals of cyclic directed graphs under any variance partition and will present some results on cyclic graphs with three and four nodes. However, we cannot guarantee that the resulting classes can be interpreted as covariance equivalence classes anymore.

5.1   Conjecture explaining computational results

Now, we want to investigate the results in Section 4.3 to analyse under which conditions two DAGs are covariance equivalent under arbitrary equal variance partitions. We want to have a closer look at when the edge between 3 and 4 is invertible under $\mathcal{P} = \{\{1, 2\}, \{3\}, \{4\}\}$ and consider the graphs (a),(b),(c) from Figure 2 again. Let $(a'), (b'), (c')$ denote the graphs that have the edge between 3 and 4 inverted. We observe the following. Graph (a) and (a') are covariance equivalent under $\mathcal{P}$ and neither (a) nor (a') has a collider in node 3 or 4. Graph (b) and (b') are covariance equivalent under $\mathcal{P}$ and both (b) and (b') have shielded or no colliders in node 3 or 4. Graph (c) and (c') are not covariance equivalent under $\mathcal{P}$ and (c) has the unshielded collider $2 \rightarrow 3 \leftarrow 4$ where (c') has no unshielded collider in 3. Motivated by these observations, we computationally verify that identical unshielded colliders are indeed a necessary condition for DAGs being in the same covariance equivalence class under $\mathcal{P}$. Putting it together, we have observed that if two graphs are in the same equivalence group under $\mathcal{P}$, they (i) have identical skeleton, (ii) identical unshielded colliders, and (iii) identical incoming and outgoing edges of nodes 1 and 2. Interestingly, these conditions are not only necessary but also sufficient to characterise the covariance equivalence classes of $\mathcal{P}$ as we can again verify computationally.

Now, let us assume that conditions (i) to (iii) are sufficient and necessary for any variance partition where we formulate condition (iii) in the more general way that incoming and outgoing edges of nodes of an equal variance group with more than one member have to be identical. With *equal variance group*, we refer to the sets in a variance partition. Then, the model uniqueness of all other partitions of Table 3 can be explained by condition (iii). Moreover, the conditions incorporate the known cases from Section 3. In the arbitrary variance case, condition (iii) becomes vacant and we precisely obtain Markov equivalence. In the case where all variances are equal, condition (iii) implies that the covariance equivalence groups consist of exactly one member. Hence, conditions (i) to (iii) bridge the gap between the two known cases and align with our computed covariance equivalence classes for any partition with three and four nodes. We make the conjecture that conditions (i) to (iii) are the precise characterisation of covariance equivalence classes for any equal error

variance partition.

Conjecture 5.1. *Let G, G' be two DAGs with n nodes and let $\mathcal{P}$ an arbitrary equal error variance partition of G. Then, G and G' are covariance equivalent under $\mathcal{P}$ if and only if G and G' have*

(i) *identical skeleton,*

(ii) *identical unshielded colliders, and*

(iii) *identical incoming and outgoing edges for nodes in equal variance groups with more than one member.*

We emphasise that the conjecture preserves the Markov equivalence classes, i.e., adding equal error variance assumptions only restricts the existing Markov equivalence classes. Finally, we remark that the conjecture is based on little empirical evidence considering that we only investigated four low-dimensional variance partitions of which only one showed groups with more than one member. Nevertheless, Conjecture 5.1 might still be promising as it reasonably generalizes the two known cases, at least for three and four nodes.

5.2 Distribution of equivalence class sizes under conjecture

Assuming that the conjecture holds true, we want to investigate the distribution of the sizes of covariance equivalence classes similar to Table 3. By property (iii) of Conjecture 5.1 and Proposition 4.13 it is clear that the distribution of the sizes of covariance equivalence classes under an equal variance partition $\mathcal{P}$ only depends on the number of vertices without any equal error variance assumption in $\mathcal{P}$. Let us call these vertices *free nodes* of $\mathcal{P}$. It is clear that one and $n-1$ free nodes is not possible, that zero free nodes implies that all models are unique, and that n free nodes correspond to the Markov equivalence classes. The class size distribution for $n \in \{4, 5, 6\}$ nodes under $2, ..., n-2$ free nodes were computed with a script that generates all DAGs with n nodes and then compares the graphs by the graphical properties of Conjecture 5.1 without involving any algebra. The script is written in Python, utilizes parallel computation and was executed on a remote machine with 128GB of memory and 32 cores in the Elastic Compute Cloud of Amazon Web Services. The script can also be found in the Github repository mentioned in Section 4.2, however, the code is inefficient and could be improved significantly. The results are shown in Table 4.

We note that in Table 4, the set of possible sizes of covariance equivalence classes under k free nodes is always identical to the set of possible sizes of Markov equivalence classes of DAGs with k nodes, cf. Table 5 generated by the same script. This seems reasonable at first glance since only the edges in between the free nodes can be changed by conditions (i) and (iii) of Conjecture 5.1 and they have to preserve unshielded colliders. However, things are more complicated since unshielded colliders and acyclicity with the non-free nodes have to be preserved as well. We are not able to give a proof but state this idea as conjecture.

n	$|D_n|$	free nodes	Number of classes	Class size distribution
4	543	2	474	(1: 85.4%), (2: 14.6%)
5	29281	2	26786	(1: 90.7%), (2: 9.3%)
		3	22025	(1: 74.3%), (2: 21.5%), (3: 3.1%), (6: 1.1%)
6	3781503	2	3554766	(1: 93.6%), (2: 6.4%)
		3	3116955	(1: 82.2%), (2: 15.8%), (3: 1.5%),(6: 0.5%)
		4	2510733	(1: 67.0%), (2: 24.8%), (3: 5.1%), (4: 1.0%), (6: 1.4%), (8: 0.4 %), (10: 0.2%), (24: 0.1%)

Table 4: Distribution of covariance equivalence class sizes for four, five, and six nodes assuming that Conjecture 5.1 holds true where the number before the colon in the tuples in the rightmost column denotes the class size and the percentage its occurrence.

n	$|D_n|$	Number of classes	Class size distribution
2	3	2	(1: 50.0%), (2: 50.0%)
3	25	11	(1: 36.4%), (2: 27.3%), (3: 27.3%),(6: 9.0%)
4	543	185	(1: 31.9%), (2: 25.9%), (3: 19.5%), (4: 10.3%), (6: 2.2%), (8: 6.5 %), (10: 3.2%), (24: 0.5%)

Table 5: Distribution of Markov equivalence class sizes with two, three, and four nodes where the number before the colon in the tuples in the rightmost column denotes the class size and the percentage its occurrence.

Conjecture 5.2. *Let $n \in \mathbb{N}$ and $k \in [n] \setminus \{1, n-1\}$. The possible sizes of covariance equivalence classes of DAGs with n nodes under a variance partition with k free nodes is equal to the possible sizes of Markov equivalence classes of DAGs with k nodes.*

Finally, we remark that the percentage of classes with one member under a fixed number of free nodes and increasing number of graph nodes seems to increase. A vague intuition could be that more non-free nodes are 'more likely' to have edges that fix edges in between the free nodes by needing to preserve unshielded colliders or acyclicity with the non-free nodes. This is far from a formal proof. Running this analysis with optimized code for higher number of nodes might give more insights.

5.3   Ideal equivalence allowing cyclic graphs

So far, we have only considered directed acyclic graphs. It is worth mentioning that we can use the methods implemented in Section 4.2 to study vanishing ideals of models from cyclic graphs by only making minor changes. However, some adjustments in their interpretation as covariance equivalence classes will need to be made. Throughout this section, let $G = (V, D)$ be any directed not necessarily acyclic graph with n nodes and no loops, that is no edges $v \to v$ with $v \in V$. We have seen in Proposition 2.1 that the

matrix $I - \Lambda$ with $\Lambda \in \mathbb{R}^D$ is always invertible for DAGs. This fails for cyclic graphs for particular values of Λ.

Example 5.3. Let $G = (\{1, 2\}, \{1 \to 2, 2 \to 1\})$. Define

$$\Lambda_1 := \begin{pmatrix} 0 & 2 \\ 1 & 0 \end{pmatrix}, \quad \Lambda_2 := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{48}$$

then $\Lambda_1, \Lambda_2 \in \mathbb{R}^D$ and $(I - \Lambda_1)$ invertible but $(I - \Lambda_2)$ is not invertible.

Following the review of Drton [3], a straight-forward solution is to define

$$\mathbb{R}^D_{reg} := \{\Lambda \in \mathbb{R}^D \mid \det(I - \Lambda) \neq 0\} \tag{49}$$

and the linear structural equation model of G under an equal variance partition $\mathcal{P}$ as

$$\mathcal{M}^{\mathcal{P}}(G) := \{(I - \Lambda)^{-T}\Omega(I - \Lambda)^{-1} \mid \Lambda \in \mathbb{R}^D_{reg}, \ \Omega \in DP^{\mathcal{P}}_n\}. \tag{50}$$

Although this seems like a minor change, the parametrization map ϕ_G is now a rational map instead of a polynomial map [3, Prop 2.1]. This makes cyclic directed graphs much more complicated, e.g. ϕ_G is not injective anymore cf. Remark 2.7. As before, we still have $\mathcal{M}^{\mathcal{P}}(G) \subset V(\mathcal{I}^{\mathcal{P}}(G)) \cap PD_n$ with a similar proof as in Proposition 3.10. However, the proof of the converse statement as in Theorem 4.1 does not hold anymore since it used that ϕ_G has a rational inverse. Finding a valid proof for the converse statement is beyond the scope of this thesis, so for now, we only know that different vanishing ideals imply different models. Hence, if the vanishing ideals of graphs are identical we will just say that the graphs are *ideal equivalent*.

We can still compute and compare the vanishing ideal in a similar fashion to Section 4.1 and Section 4.2. To apply the implictization theorem as in Section 4.1 we need a polynomial map and a variety in the domain. The inverse of $(I - \Lambda)$ is $\det(I - \Lambda)^{-1} \operatorname{adj}(I - \Lambda)$, cf. Proposition 2.1. In the acyclic case, the topological ordering implied that $\det(I - \Lambda) = 1$, however, we cannot use this fact for cyclic graphs. Since we only consider $\Lambda \in \mathbb{R}^D_{reg}$, the determinant of $(I - \Lambda)$ never vanishes and $\phi_G \cdot \det(I - \Lambda)^2$ is a polynomial map that preserves the relations among the entries of the matrices that belong to the model. Thus we can use it for implicitization. To show that $\mathbb{R}^D_{reg}$ is a variety we encode $\det(I - \Lambda) \neq 0$ with an additional indeterminate t with values in $\mathbb{R}$ and the polynomial equation

$$1 - \det(I - \Lambda) \cdot t = 0. \tag{51}$$

Since there are no additional assumptions on t, Equation (51) holds if and only if $\det(I - \Lambda) \neq 0$ and, thus, $\mathbb{R}^D_{reg} \times \mathbb{R}$ is an algebraic variety. Now, we can apply the implicitization theorem as in Section 4.1. We note that all polynomials containing t will be eliminated in the elimination ideal. We obtain

$$\mathcal{I}^{\mathcal{P}}(G) = (\langle \omega_{ii} - \omega_{jj} \mid \exists P \in \mathcal{P} : \{i, j\} \subset P \rangle + \langle 1 - \det(I - \Lambda)t \rangle + \langle \Sigma - \phi_G(\Lambda, \Omega) \rangle) \cap \mathbb{R}[\Sigma]. \tag{52}$$

Again, we eliminate Ω by substituting $K := (I - \Lambda)^T \Sigma (I - \Lambda) = (k_{ij})$. This yields

$$\mathcal{I}^{\mathcal{P}}(G) = (\langle k_{ii} - k_{jj} \mid \exists P \in \mathcal{P} : \{i, j\} \subset P \rangle + \langle 1 - \det(I - \Lambda)t \rangle + \langle k_{ij} \mid i \neq j \rangle) \cap \mathbb{R}[\Sigma]. \tag{53}$$

Instead of introducing the additional indeterminate t we can also utilize saturation.

Definition 5.4. Let $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ and $I, J \in \mathbb{K}[p]$. The *saturation* of I with respect to J is

$$I : J^{\infty} := \{f \in \mathbb{K}[p] \mid \forall g \in J \; \exists N \geq 0 : \; fg^{N} \in I\}. \tag{54}$$

If $J = \langle g \rangle$, we omit the brackets around g. Saturation yields a useful proposition.

Proposition 5.5. *[2, Thm. 4.4.14] Let $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ and $f_1, ..., f_s, g \in \mathbb{K}[p]$. Define $I := \langle f_1, ..., f_s \rangle$ and $\tilde{I} := \langle f_1, ..., f_s, 1 - tg \rangle$ where t is a new indeterminate. Then,*

$$I : g^{\infty} = \tilde{I} \cap \mathbb{K}[p]. \tag{55}$$

Readers further interested in saturation might consider the book from Cox, Little, and O'Shea [2, Chapter 4.4]. Applying this proposition to Equation (53), we finally obtain

$$\mathcal{I}^{\mathcal{P}}(G) = ((\langle k_{ii} - k_{jj} \mid \exists P \in \mathcal{P} : \{i, j\} \subset P \rangle + \langle k_{ij} \mid i \neq j \rangle) : \det(I - \Lambda)^{\infty}) \cap \mathbb{R}[\Sigma]. \tag{56}$$

To compute $\mathcal{I}^{\mathcal{P}}(G)$, we could use Equation (53), just add $1 - \det(I - \Lambda)t$ to the set $J_G^{\mathcal{P}}$ defined in Equation (46), and proceed as outlined in the end of Section 4.1. In Macaulay2 and Maple, things are even simpler since both software packages offer algorithms to compute saturations. Such an algorithm is described by Cox, Little, and O'Shea [2, p. 205]. Hence, we adjust the algorithms from Section 4.2 by only adding the saturation with $\det(I - \Lambda)$ before computing the elimination ideal and by generating all directed graphs without loops instead of only the directed acyclic graphs. The results of all variance partitions that the machine from Section 4.3 could compute with this version of the algorithm before running out of memory during elimination can be seen in Table 6. We emphasize one more time that these classes are ideal equivalence classes and not covariance equivalence classes. The number of digraphs with n nodes without loops is $4^{n(n+1)/2}$ since we have $n(n + 1)/2$ pairs of nodes v, w where there is either $v \to w$, $v \leftarrow w$, both of the edges before, or none of the edges before. We highlight that the ideal equivalence classes

n	Number of digraphs	Partition	Class size distribution
3	$4^3 = 64$	$\{\{1\}, \{2\}, \{3\}\}$	(1: 7), (2: 3), (3: 3), (42: 7)
		$\{\{1, 2\}, \{3\}\}$	(1: 64)
		$\{\{1, 2, 3\}\}$	(1: 64)
4	$4^6 = 4096$	$\{\{1\}, \{2\}, \{3\}, \{4\}\}$	(1: 59), (2: 91), (3: 40), (4: 19), (6: 4), (8: 12), (18: 6), (64: 1)

Table 6: Ideal equivalence class sizes when allowing directed graphs without loops. In each tuple in the rightmost column, the number before the colon denotes the size of the class and the number after the colon denotes the number of classes with that particular size.

of cyclic directed graphs without loops behave completely different than the covariance equivalence classes of DAGs under no equal error variance assumptions. In particular, skeletons and unshielded colliders can change within ideal equivalence classes. Moreover, we note that it seems like cyclic graphs can always change the direction of their cycles and that cyclic and acyclic graphs can belong to the same equivalence class. The ideal equivalence class under no equal error variance assumptions depicted in Figure 3 shows

most of the aforementioned behaviours. We especially emphasize the behaviour of the outgoing edges of node 4. It might be interesting to obtain results for other partitions by running the code on a more powerful machine. We remark that the Github repository mentioned in Section 4.2 contains the ideal equivalence classes of all results investigated as well as functions and documentation for further exploration.
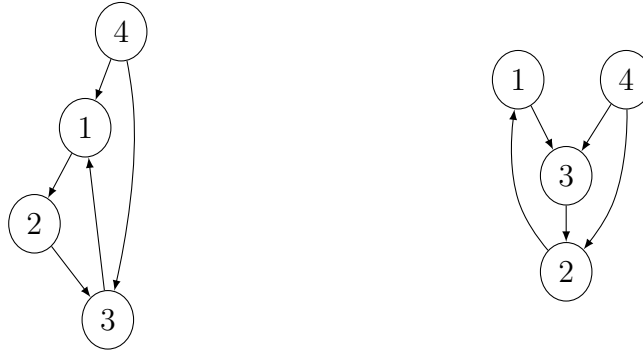

Figure 3: An interesting ideal equivalence class with two members considering directed graphs without loops with four nodes under $\mathcal{P} = \{\{1\}, \{2\}, \{3\}, \{4\}\}$.

6   Conclusion

Motivated by previous results on covariance equivalence of linear structural equation models with arbitrary [25] and equal error variances [20, 1], the goal of this thesis was to investigate covariance equivalence in the more general setting of an arbitrary set of equal error variance assumptions.

After rigorously describing the linear structural equation model with functional [3] and graphical properties [24, 5], we re-stated the known results through the new notion of equal error variance partitions. Then, we reviewed how to associate models with their vanishing ideals [3] and reviewed Gröbner basis and other computational algebra [24, 2] to give an algorithmic way to check covariance equivalence through ideal equality. We used these findings to implement an algorithm in the algebraic geometry software Macaulay2 [8] that can compute covariance equivalence classes under any variance partition and improved its performance through permutation and outsourcing of heavy computations to Maple [16]. The main result of this thesis are the covariance equivalence classes of three and four nodes under any equal variance partition computed by the algorithm. Based on these results, we stated a conjecture on covariance equivalence of graphs with arbitrary number of nodes under any equal error variance partition. The conjecture entails the two known equal error variance settings and explains the computational results on three and four nodes, cf. Conjecture 5.1. Assuming that this conjecture holds true, we also computed the possible sizes of covariance equivalence classes of graphs up to six nodes under arbitrary equal error variance partitions and stated a conjecture in the case of arbitrary number of nodes, cf. Conjecture 5.2. We remark that we could not give any proof of the two conjectures. Finally, we extended the implementation of our algorithm to also compute the equivalence of vanishing ideals of cyclic graphs [3] and presented some results for directed graphs with three and four nodes. However, these equivalence classes of vanishing ideals could note guaranteed to be identical to covariance equivalence classes. The results shown in this thesis might only be the beginning; the ready-to-use code could be further improved, cf. the thoughts at the end of Section 4.2, and executed on a more powerful machine with especially more memory capacity to obtain results on larger equal error variance partitions. These results could either solidify the conjectures or shed additional light on covariance equivalence of directed acyclic graphs under groupwise equal error variances.

References

[1] Wenyu Chen, Mathias Drton, and Y Samuel Wang. "On causal discovery with an equal-variance assumption". In: *Biometrika* 106.4 (Sept. 2019), pp. 973–980.

[2] David A. Cox, John Little, and Donal O'Shea. *Ideals, Varieties, and Algorithms.* Fourth edition. Undergraduate Texts in Mathematics. Springer-Verlag, 2015.

[3] Mathias Drton. *Algebraic Problems in Structural Equation Modeling.* 2016. arXiv: `1612.05994 [math.ST]`.

[4] Mathias Drton and Marloes H. Maathuis. *Algebraic Problems in Structural Equation Modeling.* 2016. arXiv: `1606.02359 [stat.ME]`.

[5] Mathias Drton, Bernd Sturmfels, and Seth Sullivant. *Lectures on Algebraic Statistics.* 1st ed. Oberwolfach Seminars. Birkhäuser Basel, 2009.

[6] Dan Geiger and Judea Pearl. "On the Logic of Causal Models". In: *Proceedings of the Fourth Annual Conference on Uncertainty in Artificial Intelligence.* UAI '88. 1990, pp. 3–14.

[7] Steven B. Gillispie and Michael D. Perlman. "Enumerating Markov Equivalence Classes of Acyclic Digraph Dels". In: *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence.* UAI'01. 2001, pp. 171–177.

[8] Daniel R. Grayson and Michael E. Stillman. *Macaulay2, a software system for research in algebraic geometry.* Available at `http://www.math.uiuc.edu/Macaulay2/`.

[9] Roger A. Horn. and Charles R. Johnson. *Matrix Analysis.* Second Edition. New York, USA: Cambridge University Press, 2013.

[10] Patrik Hoyer et al. "Nonlinear causal discovery with additive noise models". In: *Proceedings of the 21st International Conference on Neural Information Processing Systems.* NIPS'08. 2008, pp. 689–696.

[11] OEIS Foundation Inc. *Number of integer partitions, Entry A000041 in the on-line encyclopedia of integer sequences.* 2022. URL: `https://oeis.org/A000041`.

[12] OEIS Foundation Inc. *Number of labeled DAGs, Entry A003024 in the on-line encyclopedia of integer sequences.* 2022. URL: `https://oeis.org/A003024`.

[13] Laura Kaikkonen et al. "Bayesian Networks in Environmental Risk Assessment: A Review." In: *Integrated environmental assessment and management* 17,1 (2021), pp. 62–78.

[14] Steffen L. Lauritzen. *Graphical models.* Oxford University Press, 1996.

[15] Rusydi H. Makarim and Marc Stevens. "M4GB: An Efficient GröBner-Basis Algorithm". In: *Proceedings of the 2017 ACM International Symposium on Symbolic and Algebraic Computation.* ISSAC '17. 2017, pp. 293–300.

[16] Maplesoft, a division of Waterloo Maple Inc.. *Maple.* Version 2022. Waterloo, Ontario, 2022.

[17] Scott McLachlan et al. "Bayesian networks in healthcare: Distribution by medical condition". In: *Artificial Intelligence in Medicine* 107 (2020), p. 101912.

[18] Jelena Mojsilović, Dylan Peifer, and Sonja Petrović. *Learning a performance metric of Buchberger's algorithm*. 2021. arXiv: `2106.03676 [math.AC]`.

[19] Richard Neapolitan. *Learning Bayesian Networks*. Prentice Hall, 2003.

[20] Jonas Peters and Peter Bühlmann. "Identifiability of Gaussian structural equation models with equal error variances". In: *Biometrika* 101.1 (Nov. 2013), pp. 219–228.

[21] Thomas Richardson and Peter Spirtes. "Ancestral graph Markov models". In: *The Annals of Statistics* 30.4 (2002), pp. 962–1030.

[22] Shohei Shimizu et al. "A Linear Non-Gaussian Acyclic Model for Causal Discovery". In: *Journal of Machine Learning Research* 7.72 (2006), pp. 2003–2030.

[23] Milan Studeny. "Conditional Independence Relations Have No Finite Complete Characterization". In: *Transactions of the Prague Conferences on Information Theory 11*. 1992, pp. 377–396.

[24] Seth Sullivant. *Algebraic Statistics*. Graduate Studies in Mathematics. American Mathematical Society, 2018.

[25] Tom S. Verma and Judea Pearl. "Equivalence and synthesis of causal models". In: *Uncertainty in Artificial Intelligence 6* (1991), pp. 255–268.

[26] Jiji Zhang and Peter Spirtes. "Detection of Unfaithfulness and Robust Causal Inference". In: *Minds Machines* 18 (2008), pp. 239–271.

Appendix: Covariance equivalence classes of four nodes

We consider DAGs with four nodes and the variance partition $\mathcal{P} = \{\{1, 2\}, \{3\}, \{4\}\}$. In section Section 4.3, we have noted that the covariance equivalence classes under $\mathcal{P}$ consist of at most two graphs that have the edge between node 3 and 4 inverted. Thus, we called the edge between 3 and 4 invertible in those graphs. The following figures show all 69 graphs for which $3 \to 4$ is invertible. These figures provide enough information to deduce any covariance equivalence class under $\mathcal{P}$.
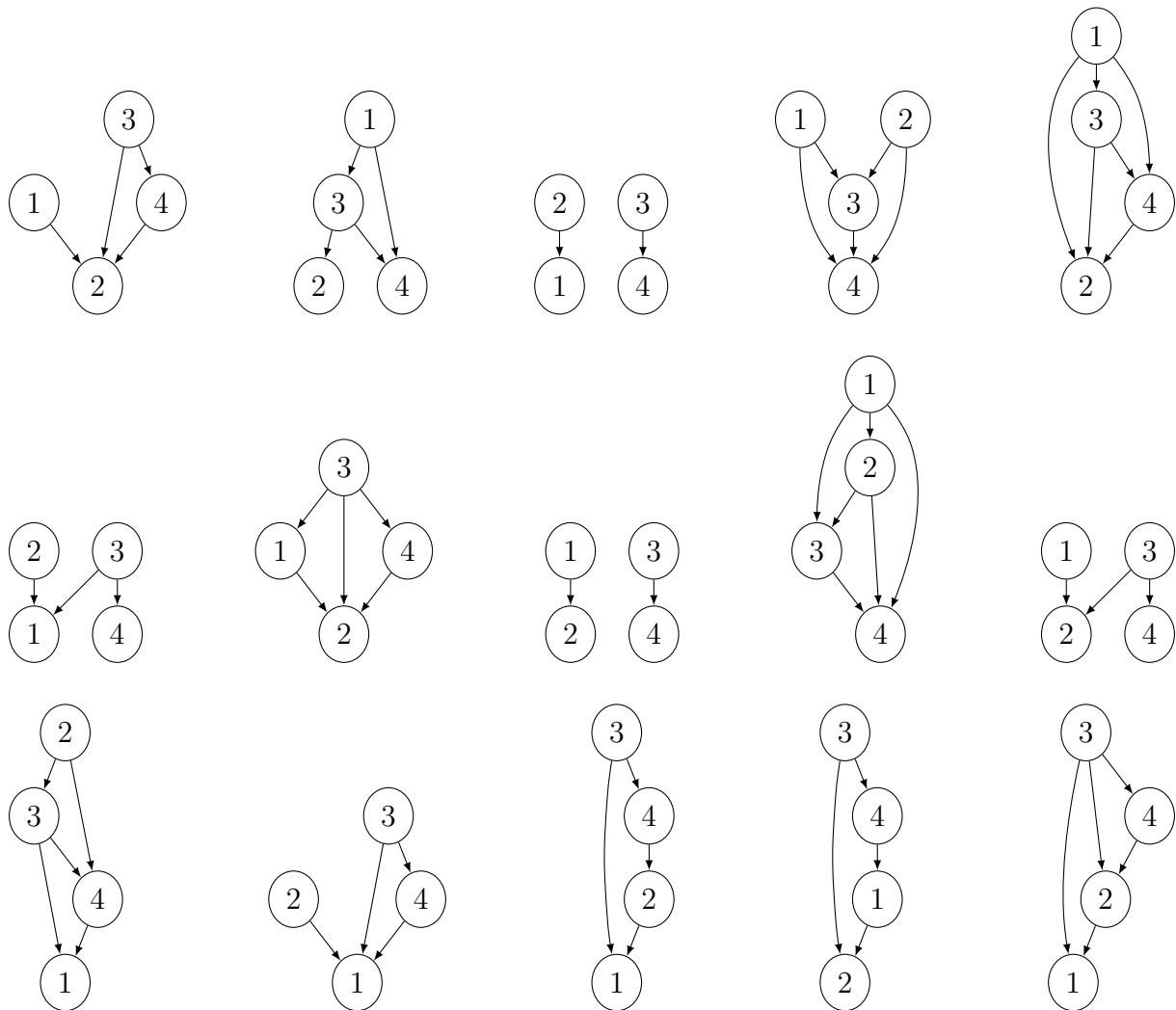
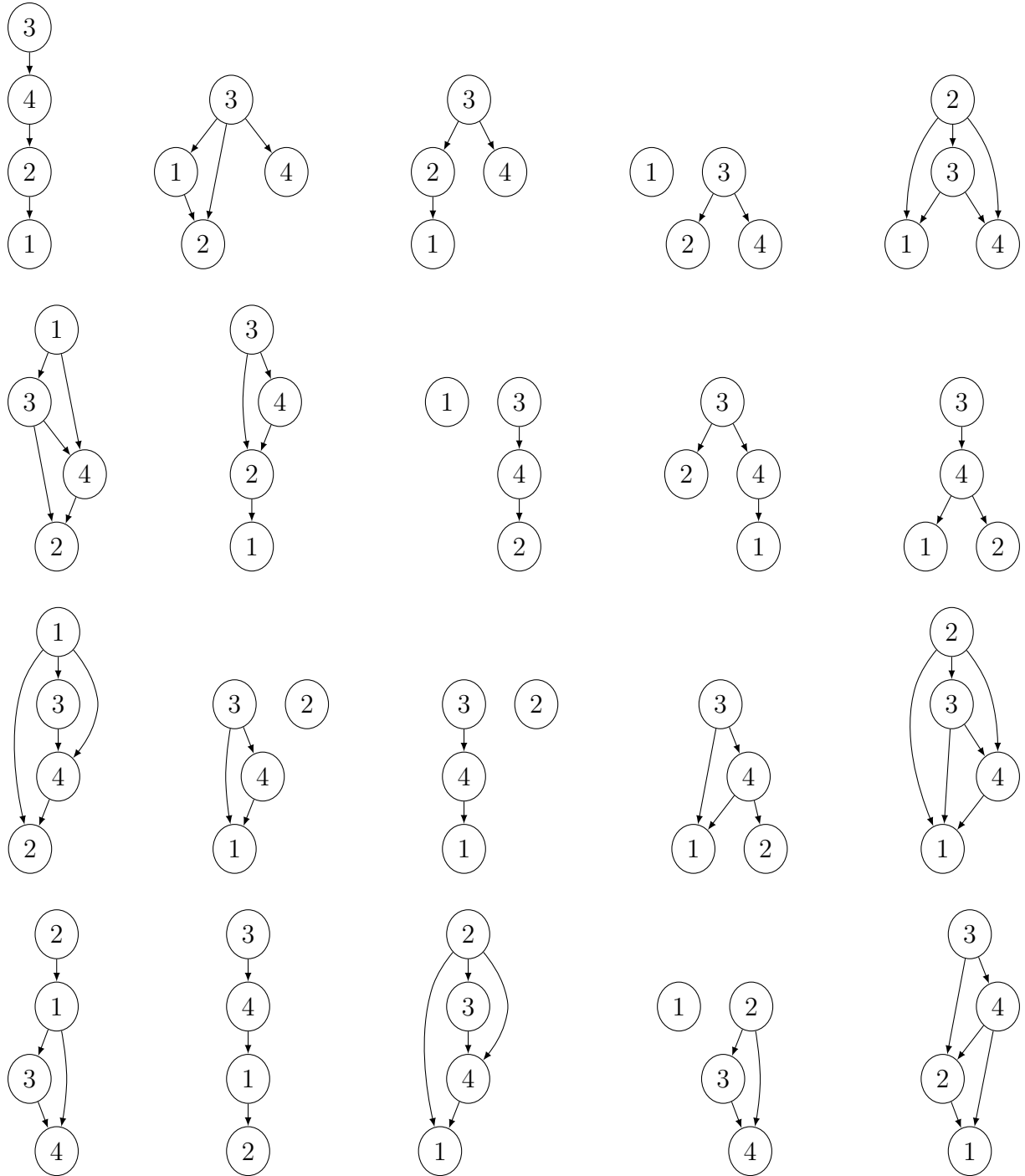

Figure 4: DAGs 1 to 15 with invertible edge $3 \to 4$

Figure 5: DAGs 16 to 35 with invertible edge $3 \rightarrow 4$
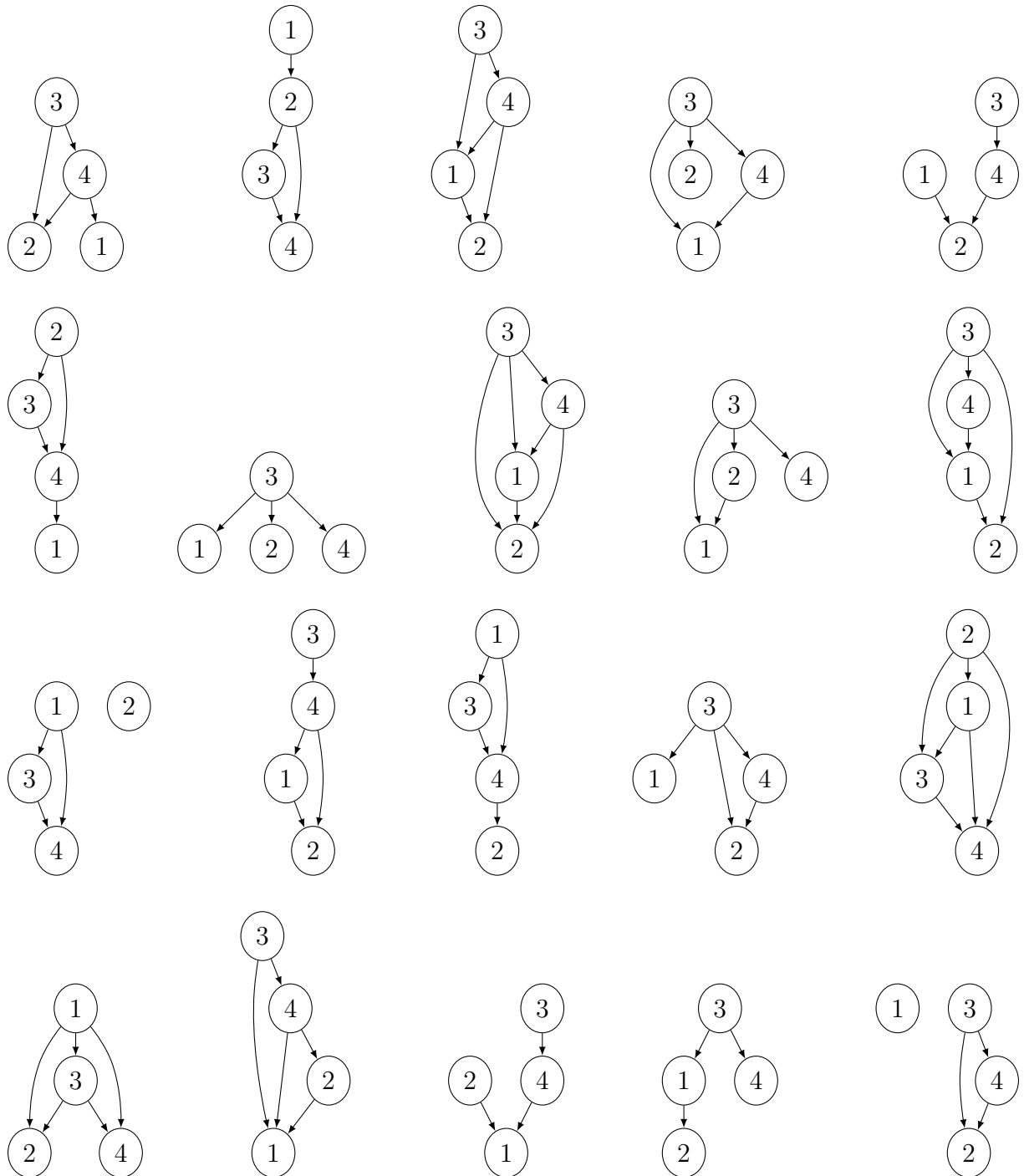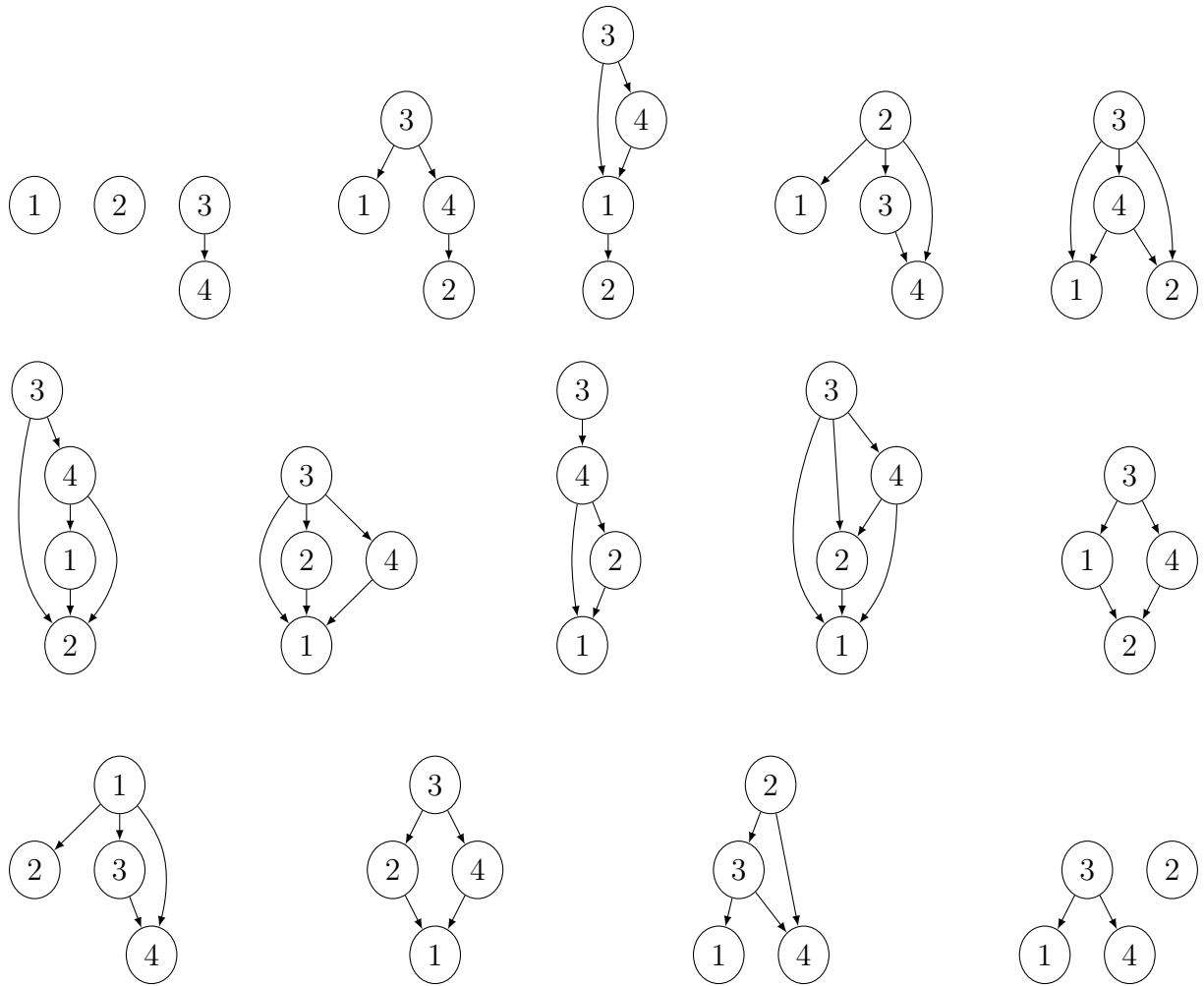
Figure 6: DAGs 36 to 55 with invertible edge $3 \rightarrow 4$

Figure 7: DAGs 56 to 69 with invertible edge $3 \rightarrow 4$